CISM International Centre for Mechanical Sciences 544
Courses and Lectures

Vincent Padois
Philippe Bidaud
Oussama Khatib *Editors*

Romansy 19 – Robot Design, Dynamics and Control

Proceedings of the 19th CISM-IFtomm Symposium

International Centre for Mechanical Sciences

Springer

CISM Courses and Lectures

Series Editors:

The Rectors
Friedrich Pfeiffer - Munich
Franz G. Rammerstorfer - Wien
Jean Salençon - Palaiseau

The Secretary General
Bernhard Schrefler - Padua

Executive Editor
Paolo Serafini - Udine

The series presents lecture notes, monographs, edited works and proceedings in the field of Mechanics, Engineering, Computer Science and Applied Mathematics.
Purpose of the series is to make known in the international scientific and technical community results obtained in some of the activities organized by CISM, the International Centre for Mechanical Sciences.

International Centre for Mechanical Sciences

Courses and Lectures Vol. 544

For further volumes:
www.springer.com/series/76

Vincent Padois · Philippe Bidaud · Oussama Khatib
Editors

Romansy 19- Robot Design, Dynamics and Control

Proceedings of the
19th CISM-Iftomm Symposium

Springer

Editors

Vincent Padois
Université Pierre Et Marie Curie

Philippe Bidaud
Université Pierre Et Marie Curie

Oussama Khatib
Stanford University

ISSN 0254-1971
ISBN 978-3-7091-1378-3 ISBN 978-3-7091-1379-0 (eBook)
DOI 10.1007/978-3-7091-1379-0
Springer Wien Heidelberg New York Dordrecht London

All contributions have been typeset by the authors
Printed in Italy

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

PREFACE

The first CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators was held on September 5-8, 1973 at CISM in Udine, Italy. The Symposium has been called RoManSy for Robot and Manipulator Systems. Indeed, RoManSy has been the very first international symposium dedicated to the Robotics field. The chairman of the first RoManSy was Professor A.E. Kobrinsky, and personalities such as Professors M. Konstantinov, I.I. Artobolevski, G. Bianchi, A. Morecki, B. Roth, M. Vukobratovic, were members of the Program Committee.

For over 35 years now, the RoManSy symposia provided a framework for fruitful exchanges between researchers around the world on modelling and design of complex robotic systems, design of control systems and interactions induced in challenging applications of robotic systems. The RoManSy has been and remains the place where scientific questions such as those related to the kinematic analysis and synthesis of complex mechanisms, the design of advanced robotic systems, the analysis of the dynamic behaviour of robotic systems and their applications in order to achieve a certain level of autonomy or adapt to changes in physical and human environment are specifically discussed.

Modelling issues of robotic systems, their physical and cognitive interactions with the environment and humans, and their dynamic control have been the core of the exchanges among attendees during RoManSy 2012. They took place based on a set of technical sessions that have involved contributions in a number of fundamental and applied aspects related the design and the control of parallel manipulators for challenging applications with a particular focus on cable-driven machines, human-robot interfaces including those for physical interactions such as those useful for physical rehabilitation, human modelling and humanoid control as well as the design of integrated perception devices, mobile robots navigation on natural terrains, etc.

Each session was introduced by a presentation placing the state of the art in the field and defining a number of scientific challenges. In

addition, the invited speakers (Professor A. Bicchi and A. Edsinger, co-founder of Meka Robotics) offered a unique perspective on particularly rich topics of research: design of bio-inspired dexterous hands and design of human friendly robots.

Future researches on the various topics covered by the 2012 RoManSy are obviously particularly important. Clearly, by considering their impact on the development of next-generation robotic systems, they will be subject to numerous investigations in the coming years and will be major topics for future editions of RoManSy to which we wish a great success.

Philippe Bidaud, Université Pierre et Marie Curie
Oussama Khatib, Stanford University
Vincent Padois, Université Pierre et Marie Curie

CONTENTS

CHAPTER VI

Humans and Humanoids

CHAPTER VII

Perception

Chapter I
Parallel Robots Modelling and Analysis

Wire-driven parallel robot: open issues

Jean-Pierre Merlet*
INRIA Sophia-Antipolis

Abstract Wire-driven parallel robot (WDPR) is a special class of parallel robot in which the rigid legs are replaced by wires, with potential advantages in terms of intrusivity and workspace. Although the study of WDPR seems to be a well-addressed subject, we will show that there are still numerous challenging open issues in this field.

1 Wire-driven parallel robots

Wire-driven parallel robot (WDPR) is a special class of parallel robot in which the rigid legs are replaced by wires. As for classical parallel robot, motion of the platform may be obtained either by changing the lengths of the wires (*type 1*) or having fixed wires lengths and modifying the location of the attachment point A of the wires on the base (*type 2*). In the first case wire lengths may be modified by using either a coiling winch or by using a linear actuator with a pulleys system (Merlet, 2010). We may also distinguish *completely restrained* robot where the wires fully constrained the n d.o.f. of the platform (in which case the number of wire must be at least $n+1$ (Ming et al., 1994)) and *cable suspended* robot with at least n wires, gravity playing the role of a virtual downward pulling wire.

WDPR have been introduced in the 80's (Landsberger and Sheridan, 1985),(Miura and Furuya, 1984) as an alternate to parallel robot with rigid links. The foreseen advantages was less intrusive legs, a simpler mechanical structure (passive joints are eliminated) and potentially larger workspace for the type 1, as the amount of leg lengths variation may be much larger than with rigid legs. WDPR shares with classical parallel robots the ability to manipulate large load and to be energy efficient. But the major difference is that wires can be pulled but not pushed, which imposes an unilateral constraint: that must be checked. We will see that this constraint greatly complexifies the analysis of WDPR.

*The author acknowledges the partial support of the EU through the grant 285404 CableBOT CP-FP

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_1,

Several prototypes have been built in the 90's, among them the famous ROBOCRANE (Albus et al., 1993), the FALCON robot (Kawamura et al., 1995) and the rescue robot of Tadokoro (Tadokoro et al., 1999), while the principle was partly patented (Thompson and Campbell, 1996). In the 2000's further prototypes have been developed such as the SEGESTA robot (Hiller et al., 2005) and other prototypes (Barrette and Gosselin, 2005),(Fattah and Agrawal, 2005).

Recently there has been a renewal of interest for WDPR in view of new applications: wind tunnel (Yaqing et al., 2007), biomechanic and rehabilitation (Wu et al., 2011), haptic interface (V. Zitzewitz et al., 2009), rescue robotics(Merlet and Daney, 2010) and telescope (Z-F et al., 2011) to name a few. Type 2 robots are illustrated in (Michael et al., 2009) in which several quadrotors are used to tow a load.

In spite of all these works it appears that many issues that have been investigated for such robots need to be revisited as they are not fully understood.

2 Kinematics

We first define the *wire configuration* of a WDPR at a pose as the set of wire numbers which are under tension. Clearly the unilateral constraint imposed on wires requires to connect kinematics and statics. Indeed, the geometrical constraint that relates the wire length ρ to the distance d between the wire anchor points on the base and platform must take into account the tension τ in the wire (with $\tau > 0$ if the wire is under tension). More precisely we have $\rho = d$ if $\tau > 0$ and $\rho \geq d$ for $\tau = 0$ i.e. the number of kinematic equations will depend upon the wire configuration. This does not impact the inverse kinematics (IK) if we consider that it provides d (or equivalently the location of A for type 2 robot). But the direct kinematics (DK) is another story. Indeed it must be noted that the sensors of the robot provide the measurement of ρ, while the pose of the platform is a solution of the IK which uses only d. If we assume that $\rho = d$ (i.e. all wire are under tension) we end up with the DK problem of classical parallel robots, which has usually several solutions. But nothing guarantee that in the current pose of the robot all wires are under tension. If we focus on a n wire spatial cable suspended robot, the IK provides m equations (for the $m \leq n$ wires under tension), $n - m$ inequalities $\rho \geq d$, while the mechanical equilibrium provides 6 equations. As the number of unknowns is $6 + m$ (the 6 pose parameters and the m τ) we always end up with a square system, whatever is the wire configuration. All possible DK solutions will be obtained by considering all the systems obtained for $m = 1 \ldots n$.

If $m = 1, 2$ the DK system can easily be solved, while for $m = 6$ the system may be decoupled into 2 sub-systems: the DK of a classical parallel robot (problem A) whose solving provides the pose parameters, and the linear system of the mechanical equilibrium that will provide the τ: the DK solutions will be obtained for the one of problem A for which the τ are positive. But the problem is much more complex for $m = 3, 4, 5$, for which there is no decoupling, and which have respectively 9, 10 and 11 equations, although it must be noted that it is possible to reduce the system to 6 equations. Indeed the mechanical equilibrium condition is equivalent to have the wires lines and the vertical line going through the center of mass of the platform spanning a linear complex, resulting in $6 - m$ geometrical conditions, which, added to the m IK equations, provide the necessary 6 equations (note however that after solving the system it is necessary to check the τ and to retain only the solutions which have positive tension).

We have recently used this approach to exhibit a solution for $m = 3$ (Carricato and Merlet, 2011). After some intensive calculation we have been able to reduce the system of 6 equations to an univariate polynomial of degree 158. But solving the DK for $m = 4, 5$ is still eluding us and this is clearly a major issue for WDPR. We have also here a practical issue regarding numerical solving: the algebraic approach apparently leads to high degree polynomial that cannot be safely numerically solved. Consequently we will have to rely on other numerical approaches. Interval analysis has been successfully used for $m = 3$, but preliminary work for $m = 4, 5$ have shown that the task was much more demanding. Real-time solving of the DK is not an issue, provided that 1) a guaranteed solving scheme is used (Merlet, 2004)), 2) the number of wire under tension does not change (see section 3). For large-scale robot other factor may influence the IK and DK such as the sagging of the wire or their elasticity (Kozak et al., 2006),(Gouttefarde et al., 2012),(Riehl et al., 2009). Stability of a pose should also be evaluated to eliminate unstable DK solutions (Bosscher and Ebert-Uphoff, 2006),(Carricato and Merlet, 2011).

Determining the current pose of the platform without a priori information on the pose is still an open issue.Adding information is necessary (e.g. measuring the wire tensions or directions of the wires) but such measurement is noisy and it is unclear how robust the calculation will be.

3 Singularities

Up to now it is considered that singularity analysis of WDPR does not differ from the one for classical parallel robots (Ottaviano and Ceccarelli, 2007). A first note is that for cable-suspended robot the mechanical equilibrium

condition is equivalent to the singularity analysis of a set of lines (with a close connection to grasping (Ebert-Uphoff and Voglewede, 2004). A second note is that the singularity of fully constrained WDPR is still an open issue. This is especially true as we have to consider that the the infinitesimal motion obtained in a singularity may possibly leads to a different wire configuration and hence to a different set of kinematic equations whose jacobian may become full rank. A companion question for cable-suspended robot is to determine the singular configuration in which the wire tension may indeed become infinite. This is a complex issue because we cannot restrict the study to a local analysis: in the vicinity of a singularity the wire configuration may change in such way that the robot will never be in the wire configuration for which the singularity has been determined.

We propose also to classify as singularity the pose at which there are multiple possible wire configuration. Indeed the control law will depend upon the current wire configuration and may thus fail if an undetected change of wire configuration occurs. Furthermore as for classical singularity the platform may gain uncontrollable d.o.f. at such pose.

4 Workspace and planning

Workspace analysis for WDPR must consider that a pose lie within the workspace if the geometrical constraints are fulfilled but also if the tension in the wires are positive. Hence the load has to be considered: it may be fixed (e.g. for cable-suspended robot), or its components may be restricted to lie within some ranges or it may be arbitrary (*wrench feasible workspace*). There have been numerous works on this subject see for example (Barrette and Gosselin, 2005), (Diao and Ma, 2008), (Gouttefarde et al., 2011), (McColl and Notash, 2011), (Riechel and Ebert-Uphoff, 2004), (Stump and Kumar, 2006), (Verhoeven, 2004). Wire interference has also been considered Merlet (2004) although interference is less damaging and may be accepted (Y. et al., 2008). But we have to extend workspace calculation to take into account singularity and possible change in wire configuration. Similarly for trajectory planning a path planner should avoid singularity (in the broad sense defined in the previous section), while it is necessary to determine in real-time if a wire configuration change may occur in the vicinity of the current pose. A further issue is to be able to detect a wire configuration change: this may obtained either by wire tension measurements and/or measurements of the wire directions. However both measurements are noisy and the detection, if any, will not occur immediately after a change in wire configuration. We will then have to design a recovery strategy to get the robot back on track and with all wires under

tension, whenever it is possible. Other criteria may be taken into account by the planner, such as energy.Clearly dimensional synthesis is also an open issue, especially as WDPR hardware may be designed in a modular way for allowing easy change in their geometry (provided an efficient communication means between the components of the WDPR).

5 Redundancy and control

Redundancy in WDPR is not a well addressed problem. From the kinematic viewpoint a WDPR is not a redundant robot as the IK has usually a single solution. It may however be thought that a WDPR is redundant from a static viewpoint, so that we can modify the tension distribution while keeping the platform at the same pose (Pott et al., 2009). Unfortunately its seems that this is not possible for cable-suspended robot with non-elastic wires such as the $N-1$ ($N \geq 4$ wires connected at the same point on the platform) as this robot will have always at most 3 wires under tension (Merlet, 2012). For completely restrained robot and non-elastic wires we have a control problem as we cannot control both the wire length (to keep the platform at the same pose) and the tension in the wires. For elastic wires the situation may be different as wire length control is basically equivalent to tension control in that case. But we still have the problem of wire configuration changes: it seems that such changes does not modify drastically the platform pose, while on the other hand large changes in the wire tensions will occur (Merlet, 2012). It appears also that small uncertainties in the wire stiffness have a small influence on the pose but a large one on tension in the wires. Hence position and velocity control should work fine while force control will be difficult Krut et al. (2004),(Oh et al., 2005) and should be robust with respect to error in the stiffness estimation (Yu et al., 2010) Clearly we have to find better ways to fully exploit the possible redundancy of WDPR. A possible approach and intriguing problem is related to the kinematics and tension distribution in multiple WDPR whose platforms and even wires may be interconnected in a flexible way by wires (with fixed lengths or variable lengths).

6 Dynamics

Dynamics of WDPR is clearly simpler than for classical parallel robots (Bruckman et al., 2008),(Korayem et al., 2010). It may even be used to increase the workspace of the robot (Barrette and Gosselin, 2005), (Gosselin et al., 2012). But an open issue is to investigate if dynamics can also be used to manage wire configuration.

7 Conclusion

Surprisingly although numerous works have been devoted to WDPR it appears that numerous issues, even fundamental one e.g. kinematics, are still not fully understood. The unilateral constraint imposed by the wire tension imposes to revisit all these topics. It greatly complexify the problems, leading to many of the more challenging contemporary problems in kinematics but is worth investigating as WDPR have a large potential for applications.

Bibliography

J. Albus, R. Bostelman, and N. Dagalakis. The NIST ROBOCRANE. *J. of Robotic Systems*, 10(5):709–724, 1993.

G. Barrette and C. Gosselin. Determination of dynamic workspace of cable-driven planar parallel mechanisms. *ASME J. of Mechanical Design*, 127 (2):242–248, 2005.

P. Bosscher and I. Ebert-Uphoff. Disturbance robustness measures for underconstrained cable-driven robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 4206–4212, Orlando, 2006.

T. Bruckman et al. *Parallel manipulators, New Developments*, chapter Wire robot part II, dynamics, control & applications, pages 133–152. ITECH, 2008.

M. Carricato and J-P. Merlet. Direct geometrico-static problem of underconstrained cable-driven parallel robots with three cables. In *IEEE Int. Conf. on Robotics and Automation*, pages 3011–3017, Shangai, 2011.

X. Diao and O. Ma. Workspace determination of general 6 d.o.f. cable manipulators. *Advanced Robotics*, 22(2-3):261–278, 2008.

I. Ebert-Uphoff and P.A. Voglewede. On the connections between cable-driven robots, parallel manipulators and grasping. In *IEEE Int. Conf. on Robotics and Automation*, pages 4521–4526, New Orleans, 2004.

A. Fattah and S.K. Agrawal. On the design of cable-suspended planar parallel robots. *ASME J. of Mechanical Design*, 127(5):1021–1028, 2005.

C.M. Gosselin, P. Ren, and S. Foucault. Dynamic trajectory planning of a two-dof cable-suspended parallel robot. In *IEEE Int. Conf. on Robotics and Automation*, pages 1476–1481, Saint Paul, 2012.

M. Gouttefarde, D. Daney, and J-P. Merlet. Interval-analysis based determination of the wrench-feasible workspace of parallel cable-driven robots. *IEEE Trans. on Robotics*, 27(1):1–13, 2011.

M. Gouttefarde et al. Simplified static analysis of large-dimension parallel cable-driven robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 2299–2305, Saint Paul, 2012.

M. Hiller et al. Design, analysis and realization of tendon-based parallel manipulators. *Mechanism and Machine Theory*, 40(4):429–445, 2005.

S. Kawamura et al. Development of an ultrahigh speed robot FALCON using wire drive system. In *IEEE Int. Conf. on Robotics and Automation*, pages 215–220, Nagoya, 1995.

M.H. Korayem, H. Tourajizadeh, and M. Bamdad. Dynamic load carrying capacity of flexible cable suspended robot: robust feedback linearization control approach. *J. of Intelligent and Robotic Systems*, 60(3-4):341–363, 2010.

K. Kozak et al. Static analysis of cable-driven manipulators with non-negligible cable mass. *IEEE Trans. on Robotics*, 22(3):425–433, 2006.

S. Krut, O. Company, and F. Pierrot. Force performance indexes for parallel mechanisms with actuation redundancy, especially for parallel wire-driven manipulators. In *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3936–3941, Sendai, 2004.

S.E. Landsberger and T.B. Sheridan. A new design for parallel link manipulator. In *Proc. Systems, Man and Cybernetics Conf.*, pages 812–814, Tucson, 1985.

D. McColl and L. Notash. Workspace formulation of planar wire-actuated parallel manipulators. *Robotica*, 2011.

J-P. Merlet. Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis. *Int. J. of Robotics Research*, 23(3):221–236, 2004.

J-P. Merlet. Analysis of the influence of wire interference on the workspace of wire robots. In *ARK*, pages 211–218, Sestri-Levante, 2004.

J-P. Merlet. MARIONET, a family of modular wire-driven parallel robots. In *ARK*, pages 53–62, Piran, 2010.

J-P. Merlet. The kinematics of the redundant n-1 wire driven parallel robot. In *IEEE Int. Conf. on Robotics and Automation*, pages 2313–2318, Saint Paul, 2012.

J-P. Merlet and D. Daney. A portable, modular parallel wire crane for rescue operations. In *IEEE Int. Conf. on Robotics and Automation*, pages 2834–2839, Anchorage, 2010.

N. Michael, J. Fink, and V. Kumar. Cooperative manipulation and transportation with aerial robots. In *Robotics: Science and Systems*, Seattle, 2009.

A. Ming, M. Kajitani, and T. Higuchi. Study on wire parallel mechanism. In *2nd Japan-France Congress on Mechatronics*, pages 667–670, Takamatsu, 1994.

K. Miura and H. Furuya. Variable geometry truss and its application to deployable truss and space crane arms. In *35th Congress of the Int. Astronautical Federation*, pages 1–9, Lausanne, 1984.

S-R. Oh et al. A dual stage planar cable robot: dynamic modeling and design of a robust controller with positive inputs. *ASME J. of Mechanical Design*, 127(4):612–620, 2005.

E. Ottaviano and M. Ceccarelli. Numerical and experimental characterization of singularity of a six-wire parallel architecture. *Robotica*, 25(3): 315–324, 2007.

A. Pott, T. Bruckmann, and L. Mikelsons. Closed-form force distribution for parallel wire robots. In *Computational Kinematics*, pages 25–34, Duisburg, 2009.

A.T. Riechel and I. Ebert-Uphoff. Force-feasible workspace analysis for underconstrained point-mass cable robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 4956–4962, New Orleans, 2004.

N. Riehl et al. Effects of non-negligible cable mass on the static behavior of large workspace cable-driven parallel mechanisms. In *IEEE Int. Conf. on Robotics and Automation*, pages 2193–2198, Kobe, 2009.

E. Stump and V. Kumar. Workspaces of cable-actuated parallel manipulators. *ASME J. of Mechanical Design*, 128(1):159–167, 2006.

S. Tadokoro et al. A portable parallel manipulator for search and rescue at large-scale urban earthquakes and an identification algorithm for the installation in unstructured environments. In *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1222–1227, Kyongju, 1999.

C.J. Thompson and P.D. Campbell. Tendon suspended platform robot, 1996. United States Patent n° 5,585,707, McDonnel Douglas Corporation.

J. V. Zitzewitz et al. A versatile wire robot concept as a haptic interface for sport simulation. In *IEEE Int. Conf. on Robotics and Automation*, pages 313–318, Kobe, 2009.

R. Verhoeven. *Analysis of the workspace of tendon-based Stewart platforms*. PhD thesis, University of Duisburg-Essen, Duisburg, 2004.

M. Wu et al. A cable-driven locomotor training system for restoration of gait in human SCI. *Gait & Posture*, 33(2):256–260, 2011.

Wischnitzer Y., N. Shvalb, and M. Shoham. Wire-driven parallel robot: permitting collisions between wires. *Int. J. of Robotics Research*, 27(9): 1007–1026, 2008.

Z. Yaqing, L. Qi, and L. Xiongwei. Initial test of a wire-driven parallel suspension system for low speed wind tunnels. In *12th IFToMM World Congress on the Theory of Machines and Mechanisms*, Besancon, 2007.

K. Yu et al. Enhanced trajectory tracking control with active lower bounded stiffness control for cable robot. In *IEEE Int. Conf. on Robotics and Automation*, pages 669–674, Anchorage, 2010.

Shao Z-F et al. Driving force analysis for the secondary adjustable system in FAST. *Robotica*, 29(6):903–915, 2011.

A New 3-DOF Translational Parallel Manipulator: Kinematics, Dynamics and Workspace Analysis.

Pavel Laryushkin[1] Victor Glazunov[2]

[1] Moscow State Textile University 'A.N. Kosygin', Moscow, Russia
[2] Mechanical Engineering Research Institute of RAS, Moscow, Russia

Abstract A translational parallel manipulator with three degrees of freedom and three kinematic chains is considered. Each kinematic chain contains five revolute joints. Kinematics, workspace, singularities and dynamics of the proposed mechanism are discussed.

1 Introduction

Since the moment when famous Clavels Delta robot was presented (Clavel, 1987), parallel manipulators with three translational degrees of freedom have attracted much attention from researchers (Ceccarelli, 2004, Gogu, 2009) and manufacturers, as it was discovered that such manipulators are very useful in many areas (Merlet, 2006). Usually, this kind of spatial mechanisms consists of base plate, moving platform (end-effector) and three symmetric kinematic chains, also called legs or limbs (Kong and Gosselin, 2007). For instance, Delta robot has three R-R-Pa-R legs and provides pure translational motion to its moving platform in three dimensions. This mechanism is widely used in packaging and pick-and-place operations because of its phenomenal speed capability and low inertia. Another variation of Delta mechanism was proposed by Tsai (Tsai and Stamper, 1996). The inverse variation of Delta was also studied by Briot (Briot et al., 2008). Another conceptual approach was presented by Wenger and Chablat (Wenger and Chablat, 2000). Their Orthoglide mechanism has three P-R-Pa-R identical kinematic chains. The moving plate of this mechanism is capable to achieve various complicated trajectories and the workspace of this robot is very close to a cube shape. All these manipulators are constructed using parallelograms, which are widely used in 3-DOF translational parallel manipulators, as a parallelogram directly constrains rotation about a certain axis. Carricato (Carricato and Parenti-Castelli, 2004) has discussed 3 R-U-R-R mechanism that could be treated as 3 R-R-R-R-R mechanism. Each

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_2,

leg prevents the moving platform from rotating around a certain axis and, as all three axes are linearly independent, this mechanism does not exhibit constraint singularities. Lee (Lee and Hervé, 2006) has presented a concept of the 3-R-R-R-R-R mechanism that is similar to one which presented in this paper.

In this paper, we present a 3-DOF translational parallel mechanism with three legs consisting of five revolute joints. We analyze its kinematics, workspace, singularities and dynamics. Singularity analysis is based on both Jacobian matrix (Gosselin and Angeles, 1990) and screw theory (Dimentberg, 1965, Glazunov, 2010) and dynamics is analyzed by Lagrange-D'Alembert principle. All obtained theoretical results are tested on a virtual model of the mechanism within MATLAB/Simulink environment. The main contribution of this paper is that the new type of the mechanism is discussed and analyzed. It is also shown that the proposed mechanism has no singularities within the workspace.

2 Structure, kinematics and workspace of 3-RRRRR translational parallel mechanism

The proposed mechanism is shown in Figure 1.

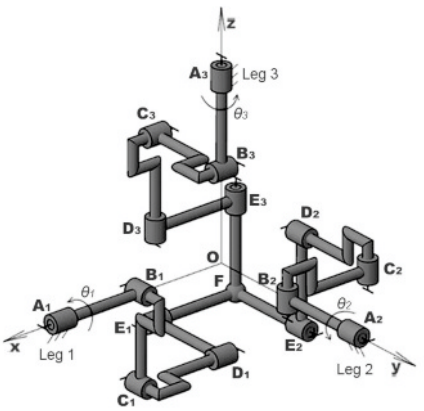


Figure 1: Kinematic scheme of proposed mechanism.

Each leg of the mechanism is constructed as follows:

– the axis of the first revolute joint of i-th (i-1, 2, 3) leg is x-, y- or z-axis (for Leg 1, Leg 2 and Leg 3, respectively) of the three dimensional Cartesian coordinate system;

- the axes of the second and the third revolute joints in each leg are orthogonal to the axis of the first revolute joint in the same leg and are parallel to each other;
- axes of the fourth and the fifth revolute joints are parallel to the axis of the first revolute joint.

Note that all three legs are symmetrical and the first R-joint in each leg is actuated. One can see that in the initial configuration of the mechanism (as shown in Figure 1) all the angles between links in each leg are right angles and following conditions must be satisfied:

$$l_4 = l_2; l_A + l_2 = l_1 + l_3 + l_5. \tag{1}$$

Here $l_1 = A_iB_i, l_2 = B_iC_i, l_3 = C_iD_i, l_4 = D_iE_i, l_5 = E_iF, l_A = OA_i$. Taking (1) into account, it was found that for the discussed mechanism a basic system of equations that represents a relationship between Cartesian coordinates x, y, z and generalized coordinates θ_1, θ_2, θ_3 can be written as follows:

$$\begin{cases} (y - l_x \sin\theta_1)^2 + (z + l_x \cos\theta_1)^2 - l_2^2 = 0; \\ (z - l_y \sin\theta_2)^2 + (x + l_y \cos\theta_2)^2 - l_2^2 = 0; \\ (x - l_z \sin\theta_3)^2 + (y + l_z \cos\theta_3)^2 - l_2^2 = 0; \end{cases} \tag{2}$$

where

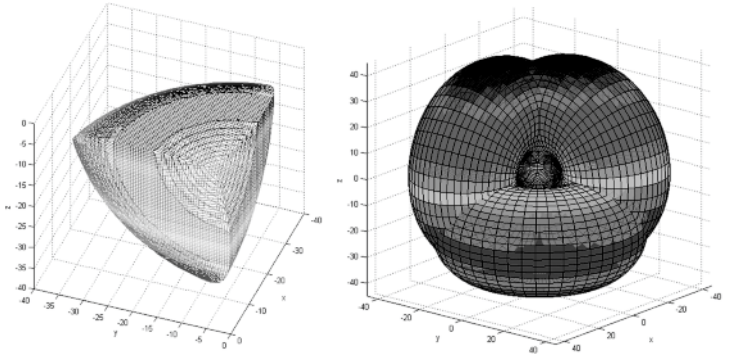
$$l_x = l_2\sqrt{1 - \left(\frac{x + l_2}{l_2}\right)^2}, l_y = l_2\sqrt{1 - \left(\frac{y + l_2}{l_2}\right)^2}, l_z = l_2\sqrt{1 - \left(\frac{z + l_2}{l_2}\right)^2}$$

This system of equations can be used to solve forward and inverse kinematic problem, i.e. finding x, y, z with given θ_1, θ_2, θ_3, and vise versa. As far as system (2) is determined, we can analyze the workspace of the mechanism by iteration method. As we can see from (2), each Cartesian coordinate can be changed only within the $[-2l_2; 0]$ interval, because exceeding this limits would result into complex roots. We can analyze all the points of the volume (a cube) restricted by these limits, and points corresponding to real number solutions of (2) will form the workspace of this robot. The result of such numerical analysis with the step equal to $0.025l_2$ between analyzed points and $l_2 = 20$ is shown in Figure 2a. This volume corresponds to an intersection of three tori with orthogonal axes (Figure 2b).

One can notice that the size of the workspace depends only on the value of l_2, as long as conditions (1) are satisfied.

3 Singularity and velocity analysis

In this section, we present results obtained after the analysis of singularities and velocities carried out by methods based on Jacobian analysis and the



(a) Workspace volume (b) Intersecting tori

Figure 2: The workspace of the mechanism

screw theory. While using Jacobian analysis method, we are assuming that each equation from system (2) could be treated as an implicit function of four variables: $F_1(x, y, z, \theta_1) = 0$, $F_2(x, y, z, \theta_2) = 0$, $F_3(x, y, z, \theta_3) = 0$ for first, second and third equation, respectively. According to Angeles and Gosselin, the relationship between input angular velocities ω_1, ω_2, ω_3 and corresponding velocities of the moving platform v_x, v_y, v_z can be written as follows:

$$\mathbf{A} \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \mathbf{B} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = 0, \tag{3}$$

where

$$\mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{x}}, \mathbf{B} = \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \tag{4}$$

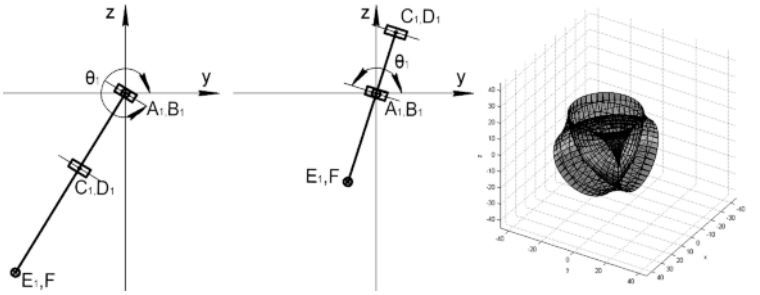
Here $\mathbf{x}$ and $\mathbf{q}$ are vectors of Cartesian and generalized coordinates, respectively. Using (3) we can easily find corresponding input velocities for any desired velocities of the moving platform and vise versa. Matrices (4) can be used for the analysis of all three types of singularities (Gosselin and Angeles, 1990). It was found that the Type 1 singularity (occurs when matrix $\mathbf{B}$ is singular) corresponds to at least one of nine following conditions:

$$x = 0; x = -2l_2; y = 0; y = -2l_2; z = 0; z = -2l_2. \tag{5}$$

$$y = -z \tan \theta_1; z = -x \tan \theta_2; x = -y \tan \theta_3. \tag{6}$$

It was also found that Type 3 singularity (when both $\mathbf{A}$ and $\mathbf{B}$ are singular) occurs when at least one of conditions (5) is satisfied. One can notice

that conditions (5) correspond to the limit points of the $[2l_2; 0]$ interval. Conditions (6) correspond to the situation when all links of any leg lie on the same plane (Figure 3a). Thus, it is easy to conclude that the conditions (6) can be satisfied only at the edge of the workspace. In Figure 3b the singular surfaces for (6) are shown.



(a) Examples of singular positions for the first leg (view from the end of the leg)

(b) Singular surfaces

Figure 3: Type 1 singularities

For Type 2 singularities (occurs if matrix **A** is singular) no analytical conditions were found using Jacobian analysis. Numerical analysis of the workspace proves that there is no point within the workspace where $det(\mathbf{A})$ is zero. Moreover, at every analyzed point $det(\mathbf{A})$ is less than zero.

We have also carried out the analysis based on the screw theory and obtained the same analytical conditions for Type 1 and Type 3 singularities, as when we used previous method. For Type 2 singularities we have found that for discussed mechanism the rank of the following matrix consisting of Plücker coordinates of wrenches (force and torque screws) must be less than six:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & W_1^0 z \\ W_1' x & 1 & W_i' z & W_1'^0 x & 1 & W_i'^0 z \\ 0 & 0 & 0 & W_2^0 x & 0 & 1 \\ W_2' x & W_2' y & 1 & W_2'^0 x & W_2'^0 y & 1 \\ 0 & 0 & 0 & 1 & W_3^0 y & 0 \\ 1 & W_3' y & W_3' z & 1 & W_3'^0 y & W_3'^0 z \end{pmatrix}$$

However, one can see that the rank of this matrix is always 6, as all the rows and columns cannot become linearly dependent. This means there is no Type 2 singular points exist for this mechanism, as we concluded it before using the Jacobian analysis. Calculations of velocities via screw

theory method have shown the same result as the calculations via Jacobian method.

The general conclusion for this section is that the proposed mechanism has no Type 2 singularities and all Type 1 and Type 3 singular points lie on the theoretical edge of the mechanism's workspace.

4 Dynamics and computer based simulation

The Lagrange-D'Alembert principle was used to analyze dynamics of the mechanism. In order to simplify the calculations, we are assuming that the masses of the links (m_1, m_2, m_3, m_4) in each leg and the mass of the moving plate m_P are distributed as shown in Figure 4

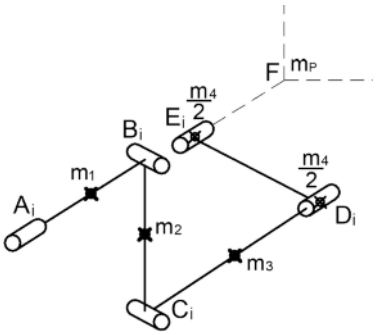


Figure 4: Distribution of the masses in a leg.

With this assumption the basic system of equations (in matrix form) which represents the dynamics of the mechanism can be written as follows:

$$J \begin{pmatrix} M_1 + M_{Fg}^1 + M_{FI}^1 \\ M_2 + M_{Fg}^2 + M_{FI}^2 \\ M_3 + M_{Fg}^3 + M_{FI}^3 \end{pmatrix} + \left(m_3 + m_P + \frac{m_2 + 4m_4}{2} \right) \begin{pmatrix} g_x - a_x \\ g_y - a_y \\ g_z - a_z \end{pmatrix} = 0 \quad (7)$$

Here M_i is a torque in actuated joint of $i-$th leg; M_{Fg}^i and M_{FI}^i are torques produced in $i-$th leg by gravity force and inertia forces, respectively; a_x, a_y, a_z are accelerations of the moving plate along axes x, y, z respectively; g_x, g_y, g_z are parts of the gravitational acceleration; J is the Jacobian matrix. Note that this system of equations represents only dynamics of the mechanism itself without taking in account any external forces and dynamics of the actuators.

In order to test theoretical dynamics model and results of the kinematics and velocity analysis, we have carried out a computer based simulation of the proposed mechanism within MATLAB/Simulink environment.

For this simulation the desired motion (position, velocity and acceleration) of the moving platform is given. Then we have solved inverse kinematics, velocity and acceleration problems obtaining angular position, velocity and acceleration of the actuated joints, and then used these parameters as input signals for the Simulink model. All motion parameters of the moving platform obtained through simulation were equal to desired motion. Thus, we can conclude that the results of the theoretical solution of the kinematics, velocity and acceleration problems are correct. As for dynamics, we have measured torques in actuated joints corresponding to the desired motion of the moving platform and then compared them with torque values obtained theoretically. In Figure 5 results of the theoretical torque calculation (gray) and computer simulation (black) are presented.

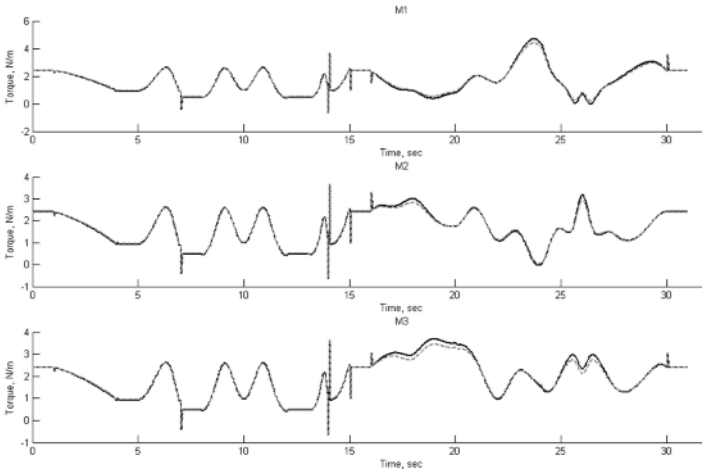


Figure 5: Torques in actuated joints.

5 Conclusion

In this paper, a 3-DOF translational parallel mechanism with five revolute joints is presented. Its kinematics, workspace, singularities and dynamics are analyzed. All the results of the theoretical analysis were tested with computer based model and were proven to be adequate. It was also shown

that the mechanism has no singular points inside its workspace but only at the edge of the workspace. Thus, the mechanism is capable of moving freely within the workspace and performing various complicated moves.

Bibliography

S. Briot, V. Arakelian, and V. Glazunov. Design and analysis of the properties of the delta inverse robot. In *Proceedings of the X. International Conference on the Theory of Machines and Mechanisms.* Liberec, Czech Republic, 2008.

M. Carricato and V. Parenti-Castelli. On the topological and geometrical synthesis and classification of translational parallel mechanisms. In *Proceedings of the 11th World Congress in Mechanism and Machine Science,* April 1-4, Tianjin. China, 2004.

M. Ceccarelli. *Fundamentals of Mechanics of Robotic Manipulations.* Kluwer Academic Publishers, 2004.

R. Clavel. Device for displacing and positioning an element in space. *Patent WO 87/03528*, 1987.

F. Dimentberg. *The Screw Calculus and its Applications in Mechanics.* Nauka. (Eng. trans.: AD680993, Clearinghouse for Federal Tech. and Sc. Inf.*, Virginia*), 1965.

V. Glazunov. Design of decoupled parallel manipulators by means of the theory of screws. *Mechanism and Machine Theory*, 45:239–250, 2010.

G. Gogu. *Structural Synthesis of Parallel Robots, Part 2: Translational Topologies with Two and Three Degrees of Freedom.* Springer, 2009.

C. Gosselin and J. Angeles. Singularity analysis of closed loop kinematic chains. *IEEE Trans. on Robotics and Automation*, 6(3):281–290, 1990.

X. Kong and C. Gosselin. *Type Synthesis of Parallel Mechanisms.* Springer, 2007.

C.-C. Lee and J. M. Hervé. Translational parallel manipulators with doubly planar limbs. *Mechanism and Machine Theory*, 24:433–455, 2006.

J.-P. Merlet. *Parallel Robots (Second Edition).* Springer, 2006.

L.W. Tsai and R. Stamper. A parallel manipulator with only translational degrees of freedom. *ASME 1996 Design Eng. Technical Conf., 96-DETC-MECH-1152*, Irvine, CA, 1996.

P. Wenger and D. Chablat. Kinematic analysis of a new parallel machine tool: The orthoglide. In *Proc. 7th International Symposium on Advances in Robot Kinematics,* Portoroz, Slovenia, 2000.

On the accuracy of $N-1$ wire-driven parallel robots

Jean-Pierre Merlet*
INRIA Sophia-Antipolis

Abstract A $N-1$ wire-driven parallel robot is a robot for which all the $N \geq 3$ wires are connected at the same point of the platform, allowing to control the location of this point. We are interested in the positioning accuracy of such a robot. If the wires are not elastic we show that the influence on the accuracy of the co-location errors of the wire anchor points on the platform is moderate, although a full analysis is a very difficult task. If the wires are elastic we study the influence of the the wire lengths measurement errors and inaccurate estimation of the stiffness of the wires. Again we show a moderate influence but very large changes in the tensions in the wires that probably prohibit the use of the redundancy to optimize the tension in the wires. In all cases the complexity of the forward kinematics of such a robot makes accuracy analysis a very demanding task that requires an in-depth investigation.

1 The $N-1$ wire-driven parallel robot

In a wire-driven parallel robot (WDPR) wires are attached at specific anchor points on the robot platform and can be coiled and uncoiled through an actuation system with a fixed output point for the wires. WDPR have been introduced in the 80's (Landsberger and Sheridan, 1985),(Miura and Furuya, 1984) as an alternate to parallel robot with rigid links. They share with them the ability to manipulate large load and to be energy efficient (Li and Bone, 2001) while they allow for larger workspace (as the amount of leg lengths variation is much larger) and present a simpler mechanical design. However their major drawback is that wires can be pulled but not pushed, which increases the complexity of their kinematics as statics has to be taken into account (for example the forward kinematic problem is an open issue (Carricato and Merlet, 2011)).

*The author acknowledges the partial support of the EU through the grant 285404 CableBOT CP-FP

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_3,

There has been recently a renewal of interest for WDPR in view of new applications: wind tunnel (Yaqing et al., 2007), biomechanic and rehabilitation (Wu et al., 2011), haptic interface (V. Zitzewitz et al., 2009), rescue robotics (Takeda et al., 2005),(Merlet and Daney, 2010). However an important point has not been addressed completely in this field: positioning accuracy. This is a very well addressed field for parallel robots with rigid links (Merlet, 2006) but still an open issue for WDPR whose kinematics is much more complex (Ottaviano et al. (2002); Thomas et al. (2002)).

We will address this problem for a specific class of WDPR, the $N-1$ WDPR which has N wires attached at the same point on the platform, allowing to control the location of this point but not the platform orientation. We will consider two different cases for the wires: non-elastic and elastic.

2 Non-elastic wires

Although this is not the scope of this paper, an important result has to be presented if $N > 3$:

At any pose a $N-1$ robot with non-elastic wires will have at most 3 wires under tension whatever is $N > 3$

This new result, that will be presented at ICRA 2012, is important as it allows to reduce the accuracy analysis of a $N-1$ robot to the accuracy analysis of the four 3-1 robots that are derived from the initial robot.

The exit point of the wire system i will be denoted by A_i, its wire length as ρ_i and the tension in the wire as τ_i. The platform pose is determined by the coordinates x, y, z of C, the center of mass of the load, in a reference frame where the $\mathbf{z}$ axis is vertical. The anchor points of the wires will be denoted by B_i (ideally $B_i = C$). Note that the inverse kinematics (IK) is straightforward as we have

$$\rho_i^2 = ||\mathbf{A_i B_i}||^2 \tag{1}$$

Let us assume that the platform is submitted to a force $\mathcal{F}$. The relation between this force and the tension in the wires is given by:

$$\mathcal{F} = \mathbf{J}^{-T}\tau \tag{2}$$

where $\mathbf{J}^{-T}$ is the transpose of the inverse jacobian matrix of the robot. The ith column J_i^{-T} of this matrix is: $J_i^{-T} = \mathbf{A_i C}^T/\rho_i$. Note that with this convention wire i is under tension if τ_i is positive. For this robot the sources of inaccuracy are errors in the wire lengths, in the location of the exit points of the wire systems and in the location of the common attachment point. Influence of the wire lengths errors is a well studied topics (Murphy,

2007),(Thomas et al., 2002) and will not be considered here. We will also assume that the location of the A_i are well known . Hence remains possible errors on the location of the B_i.

2.1 Three distinct attachment points B_i

We will first assume that the 3 wires are attached at three distinct points on the platform, that are close to C but distinct from it. To study the accuracy of the robot we will assume a given position of C and we will calculate the wire lengths with equation (1). Then we will assume that the B_i are different from C (which implies that the wire lengths affect the orientation of the platform) and solve the forward kinematics (FK). The difference between the obtained pose and the theoretical one will give us the positioning error. Unfortunately we are confronted to a major issue: the FK for a 3-3 robot is still an open problem. For the FK we have as unknowns the 6 parameters of the pose of the platform and the 3 tensions in the wires while we have 3 kinematics equations (1) and 6 statics equations (2). It has been shown that the solution(s) of this system may be calculated in theory by solving a 158th order univariate polynomial (Carricato and Merlet, 2011) but the high order of this polynomial makes the solving quite difficult. In our case we rely however on an alternate approach based on interval analysis. As an example we will consider the 3-1 robot with $A_1(0,0,0)$, $A_2(400,0,0)$, $A_3(0,400,0)$ and consider that the B_i lie on a circle of radius 5 so that $CB_1(-5,0,0)$,$CB_2(-2.5,2.5\sqrt{3},0)$, $CB_3(2.5,2.5\sqrt{3},0)$.

For C defined by (176.375, 192.375, -147.93) the theoretical wire lengths are 300, 310, 330 and the FK admits two solutions with positive tensions: (171.72, 187.24, -152.66) with the Euler angles in radian (-0.727, -0.387, -1.62) and (178.465, 194.68, -163.99) with the angles (-0.76, 0.324, 1.55). We are confronted here with a difficulty of the FK of WDPR that may admit several solutions, this increasing the complexity of the accuracy analysis. However we note the moderate positioning errors with a distance between the solutions and the theoretical one of 8.39 and 16.36. If the radius of the circle for the B_i's is reduced to 1, then we still get 2 solutions at a distance 1.99 and 2.0 from the theoretical one.

2.2 Attachment points B_i on a common ring

We will consider here that the 3 wires are connected to a ring of center U and radius r and are free to slide on this ring, although their motion must respect their initial connecting order. We will assume that the plane that includes the ring is perpendicular to the platform and that U, C lie on the same normal to the platform (figure 1).

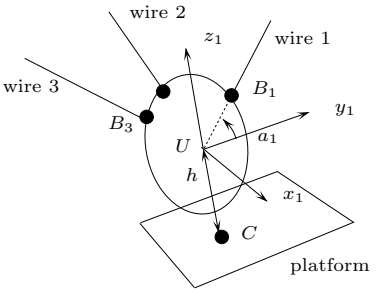


Figure 1. The wires may slide on a ring that is attached to the platform

We define a frame $\mathcal{F}_1 = U, \mathbf{x_1}, \mathbf{y_1}, \mathbf{z_1}$ attached to the ring so that $\mathbf{x}_1$ is perpendicular to the ring plane. In this frame the coordinates of B_i are

$$\mathbf{UB_i}^1 = (0, r\cos(a_i), r\sin(a_i))$$

where a_i is the angle between $\mathbf{UB_i}$ and the $\mathbf{y}_1$ axis. We then define the mobile frame of the platform $\mathcal{F}_r = (C, \mathbf{x_r}, \mathbf{y_r}, \mathbf{z_r})$ so that $\mathbf{z_r}$ is parallel to $\mathbf{z_1}$. Hence the coordinates in $\mathcal{F}_r$ of a vector $\mathbf{u}$ whose coordinates are known in $\mathcal{F}_1$ are obtained as $\mathbf{R}_1\mathbf{u}$ where $\mathbf{R}_1$ is the rotation matrix for a rotation around the axis $\mathbf{z}_r$ of angle ψ_1. As U lies on the $\mathbf{z}_r$ axis we have $\mathbf{C}U_r = (0, 0, h)$. If we define $\mathbf{R}$ as the rotation matrix between the reference frame and $\mathcal{F}_r$ we get:

$$\mathbf{CB_i} = \mathbf{RCB_i^r} = \mathbf{R}(\mathbf{CU_r} + \mathbf{R}_1\mathbf{UB_i}^1)$$

while

$$\mathbf{A_iB_i} = \mathbf{A_iO} + \mathbf{OC} + \mathbf{CB_i}$$

Equation (2) is now a linear system of 6 equations in the 3 unknowns τ_i, which is dependent upon the 3 unknowns a_i and the 6 pose parameters. Three of these equations may be used to obtain the τ_i as functions of the a_i and of the pose parameters and will remain three constraint equations.

If we assume now that there is no contact between the attachment points B_i of the wires on the ring we write that at equilibrium the force exerted by the wires at T_i must be perpendicular to the ring tangent at B_i. In the frame $\mathcal{F}_1$ the ring tangent vector $\mathbf{T}_i{}^1$ is

$$\mathbf{T}_i{}^1 = (0, -r\sin(a_i), r\cos(a_i))$$

Hence in the reference frame we have $\mathbf{T}_i = \mathbf{R}R_1\mathbf{T}_i{}^1$. We get thus 3 additional constraint equations with

$$\mathbf{A_iB_i}.\mathbf{T_i} = 0 \tag{3}$$

For the FK problem if we assume that all the B_i are distinct, then we have as unknowns the 6 pose parameters of the platform and the 3 a_i. We have also 3 equations (1), 3 remaining equations from the statics (2) and the 3 constraint equations (3). Solving such a system is quite difficult and furthermore not all solutions are valid: indeed the initial assembly of the robot imposes an ordering of the wires on the ring and this ordering must be respected for the FK solution. For solving this problem we use interval analysis which has the advantage of easily allowing to enforce the ordering constraint. For the same robot than in the previous section we have considered the pose for C as (92.95, 110.85, -198.02), which lead to the theoretical wire lengths (237.23, 376.72, 357.206), $r = 5$, $h = 10$ and the ordering (2,3,1) for the wires on the ring. The solving leads to 2 solutions for C, (110,100,200) and (110.84, 92.78, -204.22), with here again a relatively moderate difference between the theoretical pose and the final one. Note that the solving is computer intensive as the computation time is over 1h: it appears that the influence of errors on the location of the attachment points on the platform on accuracy is a very demanding task.

3 Elastic wires

In this section we assume that the wires are perfect linear springs. Let τ_i be the tension in wire i, l_i its length at rest, k_i the wire stiffness and ρ_i its length when under tension. We have

$$\tau_i = k_i(\rho_i - l_i) \tag{4}$$

Let us consider a 4-1 WDPR and its inverse kinematics. For a given pose of the load the values of the ρ_i may be determined with equation (1). Equation (2) is a linear system of 3 equations in the τ_i that allows one to calculate τ_2, τ_3, τ_4 as functions of τ_1. For a given value of τ_1 we may compute the remaining τ_i and, if all the τ_i are positive, we get the value of l_i from equation (4). The choice of τ_1 is free and hence we have a redundant robot that allows, in theory, to manage the wires tensions distribution. We may choose, for example, a τ_1 such that $H = \sum_{j=1}^{j=4} \tau_j^2$ is minimized. This function is quadratic in τ_1 and hence finding the optimal τ_1 is trivial. Note however that we may have IK solutions such that not all 4 wires are under tension.

For the accuracy analysis we have to solve the FK problem. Here the l_i are given and the pose of the load has to be determined. The first equation of (4) allows one to determine τ_1. Equation (2) is used to determine the values of τ_2, τ_3, τ_4 as functions of τ_1. The three remaining equations of (4) are linear in the coordinates of C. After solving this system we report the result in the IK equations (1) which constitutes a system of 4 equations

in the unknowns $\rho_1, \rho_2, \rho_3, \rho_4$. The difference between the first and second equation is linear in ρ_4 and is solved for this variable. The 3 remaining equations, denoted a_1, a_2, a_3, are of degree (6,6,2), (3,3,3), (9,9,3) in ρ_1, ρ_2, ρ_3. Successive resultants between these equations leads to a polynomial in ρ_1 only, which factors out in 2 polynomials of degree 76 and 96. Although this complete the theoretical solution the degree of the involved polynomials are too high to be used in practice and consequently we have to resort to a numerical procedure. For that purpose we solve the linear system (2) to get τ_2, τ_3, τ_4 as function of τ_1. Then the first equation of (4) is used to determine τ_1 as a function of ρ_1. The three remaining equations of (4) together with the 4 equations of (1) constitutes a system of 7 equations in the 7 unknowns $x, y, z, \rho_1, \rho_2, \rho_3, \rho_4$, which is solved using interval analysis. However we have also to consider that this system may not have a solution as in the final configuration less than 4 wires may be under tension. If 3 wires are under tension equation (2) is solved to determine the τ_i, the result being reported in equations (4) to obtain 3 constraint equations in x, y, z and the 3 ρ_i. With the 3 equations (1) we get a system of 6 equations in the 6 unknowns. As we have to consider all combinations of 3 wires among 4, we have to solve four such system.

To test the sensitivity of the solving to uncertainties on the l_i's and on k_i we have considered the 4-1 robot with $A_1(0,0,0)$, $A_2(400,0,0)$, $A_3(0,400,0)$, $A_4(400,400,0)$ and we have used the IK to determine what should be the l_i to reach the pose $x = 100, y = 200, z = -200$ with a load of 80, while minimizing $\sum_{j=1}^{j=4} \tau_i^2$, assuming an identical stiffness $k = 1000$ for all wires. The nominal values for the l_i are $l_1 = l_2 = 299.558$, $l_3 = l_4 = 412.108$ which leads to $\tau_1 = \tau_2 = 441.45$, $\tau_3 = \tau_4 = 202.238$. We have then considered 1000 values for the k_i that were randomly perturbed around their nominal values by $\pm 0.1k$. We have then calculated the FK by assuming first that all 4 wires were under tension and then the FK with only 3 wires under tension (all combination of 3 wires were considered) and assuming a perfect wire lengths control. For all 1000 tests we have obtained a single solution with 4 wires under tension with the ranges [99.95,100.045], [199.94, 200.05], [-200.054,-199.95] for x, y, z. However we have observed large variations in τ_i, that lie in the ranges [421.3, 462], [420.8, 461.5], [174.1,229.7],[174.7,230.4]. Furthermore there was always a single solution with wires (1,2,3) and (1,2,4) under tension, while the remaining wire is slack. For these solutions the ranges for x, y, z were [99.61,99.74], [199.71,200.28],[-200.23,-200.09]. In the triplet (1,2,3) the tension in wire 3 is almost constant (range: [402.78,403.35]) as it is for the tension in wire 4 for the triplet (1,2,4) (range: [402.78,403.35]) but the tension in wires 1, 2 were changing significantly for the two triplets with typical values of (295,588) for the first triplet and (588,295) for the

second one. Hence force measurements will allow to determine if the wires under tension are (1,2,3,4), (1,2,3) or (1,2,4).

We then perturb both l_i (with a range of ± 3) and k_i. In that case for 1000 tests, only 161 where admitting a solution with 4 wires under tension with the ranges [95.62,103.97] for x (mean value:100.18), [196.31,203.17] for y (mean value:200), [-204.22,-195.67] for z (mean value:-200.13), with large variations for the τ_i (ranges: [289.4,583.15], [290.8,590.45],[7.7, 407.2], [0.8, 402.4]). Here again there was always at least 2 solutions with 3 wires under tension with a range [94.5,104.8] for x (mean value: 99.7), [195.54,204.72] for y (mean value: 199.97), [-204.52,-195.72] for z (mean value: -200.25). For the triplet (1,2,3) τ_3 lies in the range [381.5, 426.5] while for the triplet (1,2,4) τ_4 lies in the range [383.8, 425.6]. Still force measurements allows one to determine the configuration of wire under tension as τ_1 either lies in the range [273.56,315.3] or [574.8,602.15] while τ_2 lies in the range [274.5,313.96] or [573.9,603]. Similar results were obtained for $k = 10$ and $k = 100$.

In conclusion positioning errors are moderate, while the variations of wire tensions probably prohibit the use of force control and the use of the redundancy to manage distribution of the tension in the wires.

4 Conclusion

Although WDPR have attracted a lot of interest recently there has been few works that address their positioning accuracy. We have considered a specific class of WDPR for which all wires are assumed to be attached at the same point on the platform. For non elastic wires the main source of positioning errors (beside control errors in the wire lengths) is that in practice the wires are not connected at the same point. We have shown that finding the pose of the robot when assuming close but distinct attachment points is a difficult task and seems to lead to moderate positioning errors. We have then considered WDPR with elastic wires and have shown that errors on the stiffness of the wires and/or on the wire lengths may also lead to moderate positioning errors but large variations in the wire tensions.

Accuracy analysis of WDPR is a complex task because we have to consider the FK problem for all possible combinations of wires under tension (and not only the case where all N wires are under tension), while many of these problems are open. For the $N-1$ WDPR we have shown that the positioning errors seem to be moderate but that there is large variations of wire tensions, which probably prohibit the use of force control and redundancy management. Hence this issue requires still an in-depth investigation.

Bibliography

M. Carricato and J-P. Merlet. Direct geometrico-static problem of under-constrained cable-driven parallel robots with three cables. In *IEEE Int. Conf. on Robotics and Automation*, pages 3011–3017, Shangai, May, 9-13, 2011.

S.E. Landsberger and T.B. Sheridan. A new design for parallel link manipulator. In *Proc. Systems, Man and Cybernetics Conf.*, pages 812–814, Tucson, 1985.

Y. Li and G.M. Bone. Are parallel manipulators more energy efficient ? In *IEEE Int. Symp. on Computational Intelligence in Robotics and Automation*, Banff, August 29- September 1, 2001.

J-P. Merlet. Computing the worst case accuracy of a PKM over a workspace or a trajectory. In *5th Chemnitzer Parallelkinematik Seminar*, pages 83–96, Chemnitz, April, 25-26, 2006.

J-P. Merlet and D. Daney. A portable, modular parallel wire crane for rescue operations. In *IEEE Int. Conf. on Robotics and Automation*, pages 2834–2839, Anchorage, May, 3-8, 2010.

K. Miura and H. Furuya. Variable geometry truss and its application to deployable truss and space crane arms. In *35th Congress of the Int. Astronautical Federation*, pages 1–9, Lausanne, October, 7-13, 1984.

W.S. Murphy. Determination of a position using approximate distances and trilateration. Master's thesis, Colorado School of Mines, Golden, July 2007.

E. Ottaviano et al. CaTraSys (Cassino Traking System): A wire system for experimental evaluation of robot workspace. *Robotics and Mechatronics*, 14(1):78–87, 2002.

Y. Takeda et al. A human body searching strategy using a cable-driven robot with an electromagnetic wave direction finder at major disasters. *Advanced Robotics*, 19(3):331–347, 2005.

F. Thomas et al. Uncertainty model and singularities of 3-2-1 wire-based tracking systems. In *ARK*, pages 107–116, Caldes de Malavalla, June 29- July 2, 2002.

J. V. Zitzewitz et al. A versatile wire robot concept as a haptic interface for sport simulation. In *IEEE Int. Conf. on Robotics and Automation*, pages 313–318, Kobe, May, 14-16, 2009.

M. Wu et al. A cable-driven locomotor training system for restoration of gait in human SCI. *Gait & Posture*, 33(2):256–260, February 2011.

Z. Yaqing, L. Qi, and L. Xiongwei. Initial test of a wire-driven parallel suspension system for low speed wind tunnels. In *12th IFToMM World Congress on the Theory of Machines and Mechanisms*, Besancon, June, 18-21, 2007.

Clearance and Manufacturing Errors Effects on the Accuracy of the 3-RCC Spherical Parallel Manipulators

A. Chaker[†‡], A. Mlika[†], M. A. Laribi[‡] L. Romdhane[†] and S. Zeghloul[‡]

[†] Laboratoire de Mécanique de Sousse, Université de Sousse- Ecole Nationale d'Ingénieurs de Sousse, BP 264 Sousse-Erriadh, TUNISIA

[‡] Institut PPRIME, UPR 3346, CNRS-Université de Poitiers-ENSMA, Bd Pierre et Marie Curie BP 30179, 86962 FUTUROSCOPE CHASSENEUIL, France

Abstract This paper deals with the analysis of a spherical parallel manipulator (3RCC) to determine the error on the pose of the end effector as a function of the manufacturing errors of the different links and the presence of a clearance in the joints. The obtained model allowed us to identify the error on the platform in three cases, i.e., only manufacturing errors were considered, then only clearance in the joints was considered and finally the case of both sources of error were present in the system. It was shown, in particular, that the axial displacement in the C joints is quite important. The second result is the fact that the superposition principle does not work when we consider both the manufacturing errors and the clearance despite the assumption of small displacements.

1 Introduction

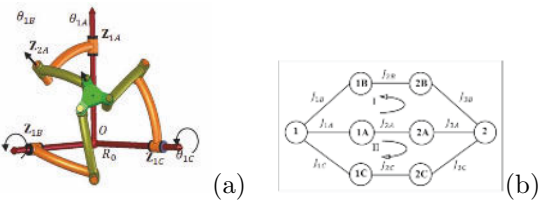
The most common architecture for SPM is the 3-RRR. However, this architecture is over constraint and some authors prefer a non overconstrained versions of the 3RRR SPM (Al-Widyan et al., 2011; Bai, 2010). The advantage of these architectures is that the assembly is always possible regardless of the manufacturing errors. But this advantage is not without affecting the desired accuracy position and orientation of the SPM. Indeed, the freedoms added to the architecture allow small displacements caused by the dimension imperfections of the links of the mechanism. Al-Widyan et al. (Al-Widyan et al., 2011) evaluated through a stochastic method the translational displacement of each cylindrical joint in the 3-RCC architecture. While these small displacements allow the mounting of the mechanism without the need to deform the links, they can yield errors on the position and orientation of the end-effector.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_4,

This problem is studied by some authors: Yu et al. (Yu et al., 2008) used a simple geometric method to analyze and compare the accuracy of three parallel robot designs. Binaud et al. (Binaud et al., 2010, 2011) proposed two aggregate sensitivity indices to compare planar parallel manipulators with regard to their workspace size and sensitivity. Clearance in the joints is another source of errors in SPM. Frisoli et al. (Frisoli et al., 2011) studied the influence of the revolute joint clearances on the position accuracy in the SPMs using a screw theory. The authors show how the position accuracy of certain parallel manipulators is strictly dependent of the mechanism pose and its association to kinematic isotropy. Tsai and Lai (Tsai and Lai, 2008) treated the joint clearances as a virtual links in a general method for error analysis of multi-loop mechanisms with joint clearances. In this work we propose to analyze the combined effect of the manufacturing errors and clearances on the pose error of the end effector of a 3-RCC SPM. The manuscript is structured as follows: after this introduction, section 2 presents the architecture of the 3RCC. Section 3 deals with the model used to calculate the errors. The obtained results are presented in Section 4, along with a discussion. Some concluding remarks are presented in Section 5.

2 Architecture of the 3-RCC SPM

Fig.1-(a) presents the architecture of the proposed spherical parallel manipulator (SPM).Three identical legs and relate the base to the platform.



(a) (b)

Figure 1. (a):The 3-RCC CAD Model, (b):The graph of the SPM

Each leg of the SPM is made out of two links and one revolute joint and two cylindrical joints. All the axes of the joints are intersecting in one point, called the center of the robot. Each link is characterized by the angle between the axes of its two joints. This angle is constant and it represents the dimension of the link. The three revolute joints with the base are actuated. This design is a non-overconstrained version of the 3-RRR

spherical parallel mechanism. The geometric parameters of one leg are the angle,ρ , between the first two joint axes and the angle,β , between the last two joint axes. The angle γ, defines the relative orientation of the axis $\mathbf{Z}_E$ of the platform and the axis of the last joint $\mathbf{Z}_{3k}$ $(k = A, B, C)$and . Since the legs are identical, the same three geometric angles are used for all the legs. A previous work (Chaker et al., 2011) was interested in the optimization of the geometric parameters in order to obtain simultaneously a large workspace with a high dexterity. This optimization, based on the genetic algorithm, provided the following results: $\rho = 39.9$, $\beta = 34.1$, $\gamma = 18.2$.

3 Model Of The Manufacturing Errors And The Clearance

In this study, we consider the manufacturing errors and clearances as small displacements that can be represented by screws of first order small displacements. We consider two types of small displacements: the first one is the given displacement, which represents the effects of the manufacturing errors of the fixed link parameters and the clearances in the kinematic joints. The second one is the unknown displacement, which represents the small variations in the joints parameters.

3.1 Formulation of the problem

For each binary link, a frame is attached to each joint as defined in section 3. We consider one of the two frames as a reference, $\Re(O, \mathbf{X}, \mathbf{Y}, \mathbf{Z})$, and the second one is $\Re'(O, \mathbf{X}', \mathbf{Y}', \mathbf{Z}')$(Fig. 2-(a)). Due to the manufacturing errors and the clearances, the reference frame is replaced by a reference frame $\Re'^{**}(O, \mathbf{X}'^{**}, \mathbf{Y}'^{**}, \mathbf{Z}'^{**})$(Fig. 2-(b)). Theoretically, the small displacement between the two links S and S' having a joint J, is described by:

$$\Im_{S'}^{S} = \Im_{S'}^{R'^{**}} + \Im_{R'^{**}}^{S} \tag{1}$$

Where $\Im_{R'^{**}}^{S}$ is a screw of small displacements describing the change between R' and R'^{**}. This screw is called *errors screw* and it is written in the frame R' as:

$$\Im_{R'^{**}}^{S} = \left\{ \begin{array}{c} \delta\alpha \\ \delta t \end{array} \right\} \tag{2}$$

Where $\delta\alpha$ is a small rotation of R'^{**} with respect to and δt a small translation from point to point.

The screw $\Im_{S'}^{R'^{**}}$ is a screw of the small variation of the joint J parameters resulting only by the manufacturing errors and clearance. This screw

is called *screw of operating error* and it has the same shape as the twist associated with the joint. For the cylindrical joint this screw is defined by

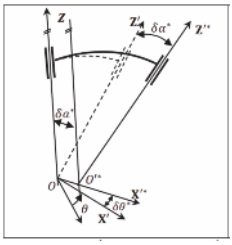
$$\Im_{S'}^{R'^{**}} = \{ \; 0 \quad 0 \quad \delta\theta \quad 0 \quad 0 \quad \delta d \; \}_{R'}^{T} \tag{3}$$

For the active joints, we assume that this operating error is zero, since these joints are locked by the actuators.

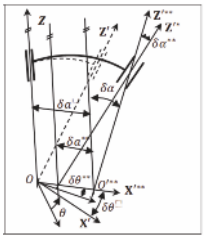
3.2 Error Screws

To express the screw given by Eq.2 we will adopt the Denavit-Hartenberg parameters. Consider a binary links having two cylindrical joints or one revolute joint and a cylindrical joint with the adjacent links. The concurrent axes **Z** and **Z'** are the axes of the two joints (Fig. 2-(a)). The angle α is a fixed parameter whereas θ is a variable parameter of the first cylindrical joint. The frame R is associated with the first joint and the frame R' is associated with the second joint.

The effect of the manufacturing errors is described by the Fig. 2-(a) and the effect of manufacturing errors and clearances is described by the Fig. 2-(b). The errors screw is made of the errors on the angle α, $\delta\alpha$, and the displacement of the center O to O'^{**} expressed in the frame R'(Fig. 2-(b)). The induced variation of the angle δ, $\delta\theta$, is not part of the screw because it corresponds to the joint variable.



(a)



(b)

Figure 2. (a):Link parameters with manufacturing errors, (b):Link parameters with manufacturing errors and clearance

When we consider only small displacements of the first order we can write

$$\delta\alpha = (\delta\alpha^{*} + \delta\alpha^{**})\mathbf{X'} \tag{4}$$

$$\delta t = (\delta a^{*} + \delta a^{**})\mathbf{X'} \tag{5}$$

3.3 The Loop Closure Equations

The two loops of the mechanism described by the following graph: Given the graph of the SPM of the Fig. 1-(b), the two loop closure equations can be written as:

• For loop I : legs A and B

$$\begin{aligned} &-(\Im_{1A}^{R_{1A}^{**}} + \Im_{R_{1A}^{**}}^{1}) - (\Im_{2A}^{R_{2A}^{**}} + \Im_{R_{2A}^{**}}^{1A}) - (\Im_{2}^{R_{3A}^{**}} + \Im_{R_{3A}^{**}}^{2A}) \\ &+(\Im_{2}^{R_{3B}^{**}} + \Im_{R_{3B}^{**}}^{2B}) + (\Im_{2B}^{R_{2B}^{**}} + \Im_{R_{2B}^{**}}^{1B}) + (\Im_{1B}^{R_{1B}^{**}} + \Im_{R_{1B}^{**}}^{1}) = \{ \begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \} \end{aligned} \tag{6}$$

• For loop II : legs A and C

$$\begin{aligned} &-(\Im_{1A}^{R_{1A}^{**}} + \Im_{R_{1A}^{**}}^{1}) - (\Im_{2A}^{R_{2A}^{**}} + \Im_{R_{2A}^{**}}^{1A}) - (\Im_{2}^{R_{3A}^{**}} + \Im_{R_{3A}^{**}}^{2A}) \\ &+(\Im_{2}^{R_{3C}^{**}} + \Im_{R_{3C}^{**}}^{2B}) + (\Im_{2C}^{R_{2C}^{**}} + \Im_{R_{2C}^{**}}^{1C}) + (\Im_{1C}^{R_{1C}^{**}} + \Im_{R_{1C}^{**}}^{1}) = \{ \begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \} \end{aligned} \tag{7}$$

Since we consider that the operating errors in the active joints are zero, the screws $\Im_{1A}^{R_{1A}^{**}}$, $\Im_{1B}^{R_{1B}^{**}}$ and $\Im_{1C}^{R_{1C}^{**}}$ are zero. These two equations yield 12 scalar equations in 12 unknown operating errors: six small rotation parameters $\delta\theta_{ik}$ and six small displacements δd_{ik}, where $i = 2, 3$ and $k = A, B, C$. The calculated axial displacement δd_{ik} can be considered as axial clearances in the revolute joints in the 3-RRR SPM designed to avoid the distortions due to the over constraint problem.

4 Results

4.1 Orientation Workspace of the SPM

Since the calculation of the error depends on the configuration of the robot, we will span the workspace of the robot by taking 100 orientations of the end-effector, given by the three Euler angles ψ, θ and φ. These 100 orientations can be represented by the location of the platform axis $\mathbf{Z}_E$, which will be located on a cone with an angle of 52°. These positions correspond to the intervals $\psi \in [12.6, 76.8]$, $\theta \in [28.8, 80.7]$and φ is taken as zero. For the manufacturing errors we chose values corresponding to a class of accuracy *f (fine)* according to the ISO 2768 norm: $\delta a = \pm 0.1mm$ and $\delta\alpha = \pm 20'$. For each one of the 100 end-effector orientations, we calculate the errors by choosing 100 arbitrary combinations of the values of the manufacturing errors. For each combination, we introduce 100 arbitrary combinations of the clearance errors. Thus we obtain in total 10^6 computation positions. We consider only the extreme values of the manufacturing errors, *i.e.*, $+0.1mm$ or $-0.1mm$ for δa and +20' or -20' for $\delta alpha$.

4.2 Calculation of the pose error

From the given screws of the manufacturing errors and the screws of operating errors calculated by equations 6 and 7, we can calculate the position δr and δO orientation errors of the end-effector as follows:

$$\begin{Bmatrix} \delta r \\ \delta O \end{Bmatrix} = \Im_2^{R^{**}_{3A}} + \Im^{2A}_{R^{**}_{3A}} + \Im^{R^{**}_{2A}}_{2A} + \Im^{1A}_{R^{**}_{2A}} + \Im^{1}_{R^{**}_{1A}} \tag{8}$$

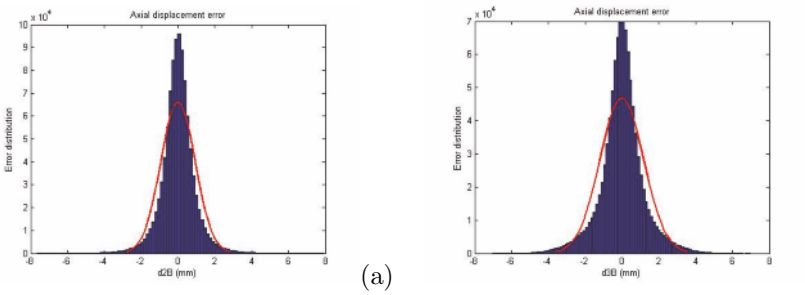
For clarity, we adopt the axis-angle representation to evaluate the error in orientation. The error angle can then be written as:

$$\delta\phi = \arccos(1/2(tr(M) - 1)) \tag{9}$$

where $M = Rot(X_0, \delta r_x) \cdot Rot(Y_0, \delta r_v) \cdot Rot(Z_0, \delta r_z)$ and δr_x, δr_v and δr_z are the components of δr.

4.3 Results and Discussion

The pose error of the end effector and the axial displacements in the C joints were calculated in three cases, i.e., only manufacturing errors were considered, then only clearance in the joints was considered and finally the case of both sources of error were present in the system. The results of the axial displacements in the C joints are given in the Table 1 and the results of the pose errors are given in Table 2. Fig. 3-(a) and Fig. 3-(b) represent examples of the distribution of the axial displacements in the C joints in case three. It is interesting to note that the distribution is binomial with a mean around 0.



(a) (b)

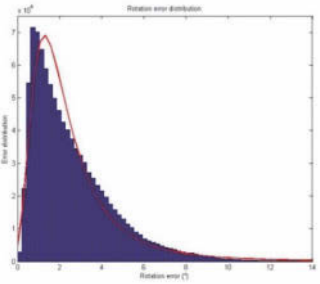
Figure 3. (a):$\delta d_2 B$ distribution, (b):$\delta d_3 B$ distribution

In the case three we obtain the distribution of the rotation error represented by the Fig. 8. This distribution is currently represented by the

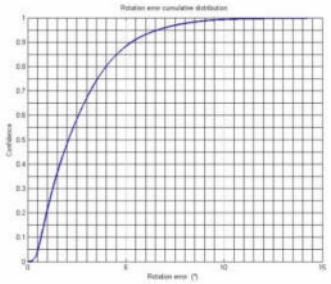
Table 1. Mean value and variance of the 6 axial displacements in the cylindrical kinematic joints

	Mean value and standard deviation (mm)		
	Without clearance	Without manufacturing errors	Manufacturing errors + clearances
δd_{2A}	0.006/0.73	0/0.6	0.016/0.945
δd_{3A}	0.003/0.78	0/0.66	0.005/1.03
δd_{2B}	0/0.7	0/0.62	-0.003/0.964
δd_{3B}	-0.009/0.92	0.001/0.75	-0.015/1.18
δd_{2C}	-0.005/0.77	0/0.64	-0.005/1
δd_{3C}	-0.004/0.68	0/0.62	-0.015/0.91

general extreme value function. Fig. 9 represents the corresponding cumulative distribution of the errors. Fig. 4-(a) represents the distribution of the rotation error. This distribution is currently represented by the general extreme value function. Fig. 4-(b) represents the corresponding cumulative distribution of the errors. At 95% of confidence we obtain the results of Table 1. Table 2 shows that the combined effect of the clearance and



(a)



(b)

Figure 4. (a):Rotation error distribution, (b):Rotation error cumulative distribution

the manufacturing errors is not the sum of these effects as one can expect. Indeed, since the hypothesis of small displacements is assumed throughout this work, one can expect that the principle of superposition is verified. However, it is clear from Table 2 that it is not the case. This fact can be explained by a sort of compensation between the errors caused by the manufacturing errors and the clearance.

Table 2. Pose error of the 3-RCC at 95% of confidence

calculated error at 95% of confidence	manufacturing errors	clearance	Manufacturing errors and clearance
Rotation error	6.5 °	1.9 °	6.62 °
Translation error	2 mm	1.66 mm	2.56 mm

5 Conclusion

In this paper, we investigated the effect of the manufacturing errors and the clearance in the joints on the pose error of the end effector. The 3RCC spherical manipulator was taken as an example. This structure is a non over constraint version of the 3RRR spherical manipulator. The axial displacement in the second and the third R joint were released to avoid the over constraint in the system. The pose error of the end effector and the axial displacements in the C joints were calculated in three cases, i.e., only manufacturing errors were considered, then only clearance in the joints was considered and finally the case of both sources of error were present in the system. The obtained results showed that the axial displacement in the C joints can be as high as $1mm$ when both sources of error are combined. This result justifies the use of the C joint instead of the R joint. The second result is the fact that the superposition principle does not work when we consider both the manufacturing errors and the clearance despite the assumption of small displacements.

Bibliography

K. Al-Widyan, X. Ma X, and J. Angeles. The robust design of parallel spherical robots. *Mech. and Mach. Th.*, 46:335–343, 2011.

S. Bai. Optimum design of spherical parallel manipulators. *Mech. and Mach. Th.*, 45:200–211, 2010.

N. Binaud, S. Caro, and P. Wenger. Sensitivity comparison of planar parallel manipulators. *Mech. and Mach. Th.*, 45:1477–1490, 2010.

N. Binaud, S. Caro, and P. Wenger. Comparaison of 3-rpr planar parallel manipulators with regard to their kinetostatic performance and sensitivity to geometric uncertainties. *Meccanica*, 46:75–88, 2011.

A. Chaker, M. A. Laribi, L. Romdhane, and S. Zeghloul. Synthesis of spherical manipulator for dexterous medical task. In *2nd IFToMM Int. Symposium on Robotics and Mechatronics,ISRM2011*, November3-5 2011. Shanghai, China.

A. Frisoli, M. Solazzi, and M. Bergamasco. A new screw theory method for estimation of position accuracy in spatial parallel manipulators with revolute joint clearances. *Mech. and Mach. Th.*, 46:1929–1949, 2011.

M. Tsai and T. Lai. Accuracy analysis of a multi-loop linkage with joint clearances. *Mech. and Mach. Th.*, 43:1141–1157, 2008.

A. Yu, I. A. Bonev, and P. Zsombor-Murray. Geometric approach of the accuracy analysis of a class of 3-dof planar parallel robots. *Mech. and Mach. Th.*, 43:364–375, 2008.

Structural synthesis of the families of parallel manipulators with 3 degrees of freedom

J. I. Ibarreche[†], O. Altuzarra, V. Petuya, A. Hernández, C. Pinto[‡*]

[†] Aernnova Engineering, Vitoria, Spain

[‡] University of the Basque Country UPV/EHU, Bilbao, Spain

Abstract The object of this article is to obtain in a systematic form all the families of parallel manipulators with 3 degrees of freedom considering solely fully parallel manipulators whose legs are structurally identical. The motion pattern of the end-effector that should be taken into account are: 3T, 2T1R, 1T2R and 3R, where T and R refer to the character of translational and rotational degrees of freedom.The methodology that follows utilizes the concepts of the theory of groups of displacements applied to the structural synthesis of parallel manipulators.

1 Introduction

In order to design a parallel manipulator (PM hereafter) for a specific application, the first step is to identify the requirements of the operation from the customer's specifications. Within these requirements is the so-called motion pattern of the platform Altuzarra et al. (2009). By such means it is understood the number and type of degree of freedom (DoF hereafter) of the platform, this is, if they are rotational or translational, if they are instantaneous or permanent, including also its directions. As from here a structural synthesis can be realized (also called morphologic) whose objective is to determine the topology of the PM, so that its mobile platform adjusts to a determined motion pattern.

There are several methods that can be used to perform the structural synthesis. One of them is the theory of groups of displacements, which uses the mathematical properties of the Lie Group which possess the displacements of rigid body. These mathematical concepts were not introduced into the field of structural synthesis of mechanisms until the initial works

[*]The authors of this paper wish to acknowledge the finance received from the Spanish Government (Projects DPI2008-00159 and DPI2011-22955), the ERDF of the European Union and the Government of the Basque Country (Project GIC07/78, IT445-10).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_5,

of Hervé (1978). In this way, the kinematics of rigid body arises from the general group of displacements of rigid body of dimension 6 and its 12 subgroups. One of the concepts used is the kinematic bonds $\mathcal{L}(i, j)$ between two elements i and j, which are defined as the relative displacement between these elements. Each one of the kinematic chains or legs of the PM generates in its end-effector a displacement which must be compatible with the pattern of movement required. The development of the different kinematic bonds, as well as a definition of their possible materializations through kinematic joints is a fundamental aspect in the structural synthesis.

As from the previous concepts, the intention is to obtain in a systematic manner, all the families of parallel manipulators of 3 degrees of freedom. Only consideration will be given to fully parallel manipulators whose legs are structurally equal, being located in a preferable symmetrical form. The motion patterns of the end-effector to be studied are 3T, 2T1R, 1T2R and 3R, where T and R refer to the character of translation and rotation of the DoF. The methodology that follows is basically presented in Salgado (2008).

The resulting architectures will be identified as existing, in case they have been referred to in previous works, and to the contrary, novel cases. In this way, the latter could be a source of new designs.

2 Essentials and Premises of the Synthesis

In order to systematize the study of the PM of 3 DoF, the following premises have to be taken into account:

1. Each kinematics chain must have at least the same number and type of DoF as the terminal element of the PM. If the kinematic chains have 6 DoF, the intersection of their motion patterns does not introduce any restrictions by which the number of the DoF of the PM will continue to be 6. Therefore the number of DoF of kinematic chains will be a minimum of 3 and a maximum of 5.
2. The structure of the kinematic chains are obtained by combining the kinematic pairs needed to obtain the number and type of DoF required. Specifically, the kinematic pairs to be used will be: of 1 DoF (Revolute R, Prismatic P), 2 DoF (Cylindrical C, Universal joint U) and 3 DoF (Spherical S).
3. The motion pattern that results in the end-effector of the chain is the product of the subgroups of displacement that compose the kinematic bonds of that chain. In other words, the product of the matrices that they represent. If there are no kinematical pairs redundant in the chain, the number of DoF of the end-effector, is the sum of the DoF of the component pairs that compose it.

4. Geometric aspects and nomenclature. It is necessary to bear in mind the geometry of the previous kinematic chains. For example, if the chain RRR has three parallel axes the generated displacement is 2T1R, if there are only two, the displacement will be 1T2R. A chain RRRRR with all its axes parallel generates in the end-effector only 3DoF with a displacement 2T1R. Nevertheless, if 3 pairs R are parallel between themselves and the other two also are between themselves and independent from the previous ones, the displacement is 3T2R. For this reason, on naming the chain for its pairs, if the axes of the pairs that compose it are parallel they will be indicated by underlining or over lining (for example $\underline{RRR}\overline{RR}$ for the previous case). To indicate that the axes cut, they will be put into parentheses (for example $3RR(RRR)$ in the case of 2T3R). Finally, the acting pair are also $\underline{\overline{\text{underlined}}}$ (double-underlined in case of coinciding with another underline).
5. The manipulators constructed as from legs with kinematic bonds of dimension 3 should have 6 redundant constraints to ensure the mobility desired in the platform. The constructed as from legs with kinematic bonds of dimension 4 should have 3 redundant constraints to guarantee the mobility desired in the platform. The constructed as from legs with kinematic bonds of dimension 5 should not have redundant constraints to guarantee the desired mobility in the platform. All of this can be seen clearly if the criteria of Grübler is applied to the different examples.
6. Finally, in this work sole consideration will be given to kinematic chains with the same dimension as its corresponding kinematic bond. That is to say that no redundant pairs will be considered as in case of the $\underline{RRRR}$ chain corresponding to the motion pattern 2T1R.

3 Generating Three DoF Parallel Manipulators

Once obtained all the possible kinematic chain generators from displacements of 3,4 and 5 DoF that can be employed Salgado (2008), it is necessary to combine them in a way that obtains the desired motion pattern in the PM of 3 DoF. All possible motion patterns to be considered in the structural synthesis are the following: 3T0R, 2T1R, 1T2R and 0T3R. The steps to follow are:

1. Selection of the kinematic chains that include at least the motion pattern considered at that time.
2. Carrying out the configuration of 3 identical kinematic chains with a geometrical disposition that the intersection of the motion pattern of the 3 chains resulted to be the sought for answer.

This work will obviate the planar PMs considering them as sufficiently studied in the bibliographies. In the figures shown below appear drawn patterns of movement of the platform according to the following criteria: double-headed arrows represent the directions of the axes of the DoF of rotation and single arrows the translation directions.

3.1 3T0R Displacements

This type of PM is known in specialized literature by its acronym in English, TPM (Translational Parallel Manipulator). Only the kinematic bonds with displacements containing at least the three translations permit obtaining this type of PM.

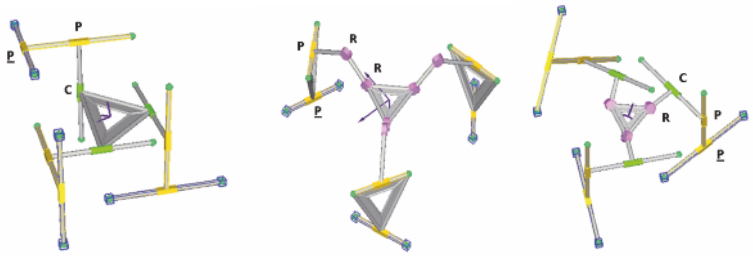
3T0R Displacements with 3 dimensional bonds. In this case the kinematic chains are PPP, PPP_a, PP_a^2, $P_a^2P_a$ and their permutations, requiring that the three directions of translation are linearly independent. A typical case is that of the PM 3- PPP Gogu (2004). Any geometrical disposition of the pairs and limbs will generate 6 redundant restrictions.

3T0R Displacements with 4 dimensional bonds. The only kinematic bond of dimension 4, which includes three translations, is the generator group of the displacements 3T1R, known as Schönflies motion or SCARA. The kinematic chains to be considered are:

- Chains PPPR, PPPaR, PP_a^2R, P_a^2PaR, PPC, PPaC, P_a^2C and their permutations. The three directions of translation have to be linearly independent.
- Chains PPRR, PPaRR, P_a^2RR, PCR, PaCR and their permutations. The revolute pairs have to be of parallel axes not coincidental, and the generated translation has to be linearly independent of the directions of the other generators of translation, and between them.
- Chains PRRR, PaRRR, CRR and its permutations. The pairs of rotation have to be non coincidental parallel axes and the direction of translation cannot be parallel to a plane perpendicular to the axes of the pairs R.

The PM originated from chains with three translation generators and one of rotation (PPPR, PPPaR,...) do not contribute any advantage with regard to the PM $3-\underline{P}PP$. Only the mechanism $3-\underline{P}PC$, and its combinations with pairs Pa, present utility (See Figure 1a).

As more interesting configurations originated from chains with two generators of translation and two of rotation we can highlight the $3-\overline{R}\underline{P}\overline{C}$ and the $3-\underline{P}RC$ proposed by Huang and Li (2004), and the PM $3\underline{P}P\overline{R}\overline{R}$ in the Figure 1b. Within the chains that have a generator of translation and



(a) $3\underline{P}PC$ (b) $3\underline{P}P\overline{RR}$ (c) $3\underline{P}PCR$

Figure 1: 3T0R Displacements generators

three rotational we can highlight the robot from the University of Maryland with a configuration $3-\overline{\underline{R}R}PaR$ and the robot $3-\overline{\underline{C}RR}$ developed almost simultaneously at the universities of Laval (Kong and Gosselin) and California Riverside (Kim and Tsai).

3T0R Displacements with 5 dimensional bonds. The only kinematic bond of dimension 5, which includes three translations is the generator group of displacement 3T2R. The kinematic chains to consider are:

- Chains,PPPRR, PPPaRR, PP_a^2RR, P_a^2PaRR, PPCR, PPaCR, P_a^2CR, PPPU, PPPaU, PP_a^2U, P_a^2PaU, CCP, PaCC and their permutations. The three directions of translation have to be linearly independent and the axes of the two rotation generators will be also independent.
- Chains PPRRR, PaPRRR, P_a^2RRR, PPUR, PaPUR, P_a^2UR, PCRR, PaCRR, PCU, PaCU, CCR and their permutations. Two translations have to be linearly independent, two of the axes of rotation have to be parallel not coincidental and independent from two translations and the third rotation.
- Chains PRRRR, PaRRRR, PRRU, PaRRU, PUU, PaUU, CRRR, CRU and their permutations. Three of the axes of rotation have to be parallel not coincidental and linearly independent of the direction of translation and of the axes of the fourth rotation; or that the rotation axes of rotation are parallel not coincidental two to two and linearly independent between them and with the translation axis.
- Chains RRRRR, RRRU, RUU and their permutations. Three rotation axes have to be parallel to each other not coincidental and also the other two as well; being both groups linearly independent groups one with other.

The kinematic chains of three translation generators and two of rotation of the type PPPRR, PPPaRR,... do not contribute anything with regard to the PM $3-\underline{P}PP$. To obtain three translations exclusively, the three equal kinematic chains of asymmetric form must be positioned in a manner so that the two rotations are annulled. This is the case of the PM $3-\underline{P}PCR$ in Figure 1c.

The configurations with two generators of translation do not give anything additional to that already studied and its constructive complexity is increased by having a greater number of kinematic pairs and elements. Worth mentioning apart are the configurations with five generators of rotation. They offer a good alternative for chains with prismatic pairs (e.g., $3-\underline{RRR}\overline{RR}$.

3.2 2T1R Displacements

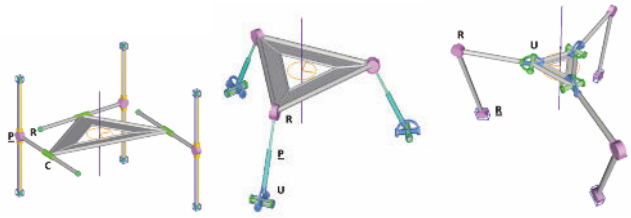
Following the applied methodology and with the premises of the design chosen at the beginning, it is not possible to find architectures with a different motion pattern than the planar motion.

3.3 1T2R Displacements

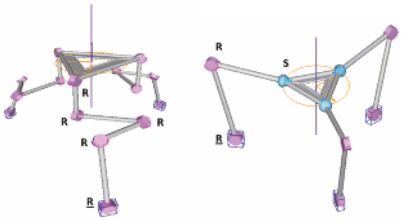
Following the applied methodology, it is understood that only the kinematic bonds with displacements containing at least one translation and two rotations will permit the obtaining of parallel manipulators with displacements 1T2R. However, those kinematic bonds containing three generators of translations should be discarded. The reason is that the intersection of the displacements of the three legs in the platform does not cancel any of the translations. This statement is not true in the case of rotations; nor is it in the case of the kinematic bond of the leg having only 2 translations or only 1.

1T2R Displacements with 3 dimensional bonds. With these chains it has not been possible to realize any construction of use

1T2R Displacements with 4 dimensional bonds. In this case, kinematic bonds to be considered are those corresponding to the displacements 2T2R and 1T3R. Some architectural platforms with displacements 1T2R are: the PM $3-\underline{P}RC$ (Figure 2a), the PM $3-U\underline{P}R$ (Figure 2b) and the PM $3-\overline{R}\overline{R}U$ (Figure 2c). With chains 1T3R has not been possible to carry out any construction of utility.



(a) $3\underline{P}RC$ (b) $3U\underline{P}R$ (c) $3\underline{\overline{R}\overline{R}}U$



(d) $3\,(\underline{R}RR)\,\overline{RR}$ (e) $3\underline{\overline{R}\overline{R}}S$

Figure 2: 1T2R Displacements generators

1T2R Displacements with 5 dimensional bonds. For the same reason above, we discard the kinematic bond 3T2R and only study those displacements 2T3R. This group belongs to the well-known PM $3-R\underline{P}S$, studied for example in Han and Tsai (2003). Other manipulators like MP $3-\underline{RR}PU$ or MP $3-P\underline{RR}U$ studied by Li and Hervé (2010) have passive DoF so are not considered here. The manipulator $3-(\underline{R}RR)\overline{RR}$ and $3-\underline{\overline{R}\overline{R}}S$ represented in Figures 2d and 2e respectively, include kinematic chains with five axes of rotation. Although in this case there is no need to define redundant constraints in order to get 3 DoF in the platform, given that kinematic bonds are of dimension 5, there is a need to impose a geometric condition in order to achieve converting one of the rotations into one translation. We must bear in mind that, except in the case where the legs have 3 translations, the first thing that the intersection of movements eliminates in the platform, are the translations.

3.4 0T3R Displacements

Only the kinematic bonds containing at least three rotations permit the obtaining of PM with displacements 0T3R. The procedure to follow is as

before.

4 Conclusions

This work has obtained a series of architectures of parallel manipulators of 3 DoF with the following motion patterns: 3T0R, 2T1R, 1T2R and OT3R. The methodology used is based on the theory of groups of displacements. Nevertheless, the relevance of the procedure has been to establish the bases and premises of the synthesis that appear in the point 2 of this article. All of them constitute a body hierarchical doctrine in a series of rules that arrange the kinematic structure of the mechanism, as well as certain conditionings on the geometric characteristics of the elements and kinematic pairs that compose the legs of the parallel manipulator. The result of the process has been the generation of a series of PM, some of which were already referenced in the literature, and other innovations that might be a source of new designs.

Bibliography

O. Altuzarra, Y. San Martín, E. Amezua, and A. Hernández. Motion pattern analysis of parallel kinematic machines: A case study. *Robotics and Computer-Integrated Manufacturing*, 25:432–440, 2009.

G. Gogu. Structural synthesis of fully-isotropic translational parallel robots via theory of linear transformations. *European Journal of Mechanics A/Solids*, 23:1021–1039, 2004.

K. Han and L-W Tsai. Kinematic synthesis of a spatial 3-rps parallel manipulator. *Journal of Mechanical Design*, 125:92–97, 2003.

J. Hervé. Analyse structurelle des mécanismes par groupe des déplacements. *Mechanism and Machine Theory*, 13:437–450, 1978.

Z. Huang and Q. Li. Type synthesis of symmetrical lower mobility parallel mechanisms using the constraint synthesis method. *IEEE Transactions on Robotics and Automation*, 20:173–180, 2004.

Q. Li and J. Hervé. 1t2r parallel mechanisms without parasitic motion. *IEEE Transactions on Robotics*, 26:401–410, 2010.

O. Salgado. *Análisis y Diseño de Manipuladores Paralelos de Baja Movilidad*. PhD thesis, University of the Basque Country UPV/EHU, 2008.

Inverse Static Analysis of Massive Parallel Arrays of Three-State Actuators via Artificial Intelligence

Felix Pasila[*§], Rocco Vertechy[†],
Giovanni Berselli[‡] and Vincenzo Parenti Castelli[*]

[*] Dept. of Mech. Eng., University of Bologna, Italy
[§] Dept. of Elec. Eng., Petra Christian University, Indonesia
[†] Percro Laboratory, Scuola Superiore Sant'Anna, Pisa, Italy
[‡] Dept. of Mech. Eng., University of Modena and Reggio Emilia, Italy

Abstract Massive parallel arrays of discrete actuators are force-regulated robots that undergo continuous motions despite being commanded through a large but finite number of states only. Real-time control of such systems requires fast and efficient methods for solving their inverse static analysis, which is a challenging problem. Artificial intelligence methods are investigated here for the on-line computation of the inverse static analysis of a planar parallel array featuring eight three-state force actuators and possessing one degree of revolute motion.

1 Introduction

Discrete-State Manipulators (DSM) are a special kind of mechanisms whose actuators can be made switching among a finite number of states only. Introduced in the early 1970's [1] in an attempt to conceive sensor-less robots as well as to reduce the complexity of control systems and computer interfacing, nowadays DSM can be classified into two different groups depending on whether their actuators act as either discrete displacement generators [2-5] or discrete force generators [6]. This work deals with the latter type of DSM, usually referred to as Massively Parallel Robots (MPR). In essence, MPR are dynamically constrained mechanisms employing a large number of on-off actuators that exert either a constant force (active state) or no force (inactive state) irrespective of their arbitrary kinematically unconstrained configuration. To achieve high force capabilities (both in terms of variation range and accuracy), the architecture of these MPR practically requires a large number of actuators (typically 4-10 times greater than the number

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_6,

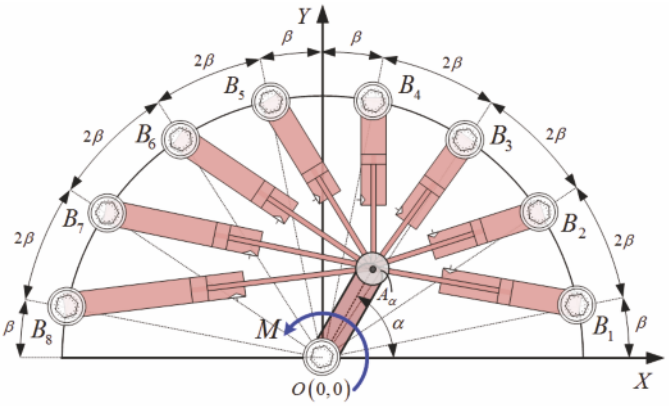


Figure 1. Ternary Massively Parallel Robot (MPR) actuated by eight three-state force generators.

of degrees of freedom desired for the robot) that are usually arranged in a prevalently in-parallel configuration. Owing to the large number and the discrete nature of the actuator variables, the Inverse Static Analysis (ISA) of MPR, i.e. to find the states of the actuator variables for a given external (force/torque) action, turns out to be a very challenging problem. To tackle this issue, in the last twenty years several elegant solution methods have been proposed [2-6], which however require too many calculations for on-line MPR control. In this paper, the potentialities of artificial intelligence methods are investigated for the real-time ISA solution of a planar ternary (i.e. with three states being -1, 0 and 1) MPR with one degree of rotational motion that is actuated by eight three-state force generators.

2 Ternary Massively Parallel Mechanism

The ternary MPR considered in this study is depicted in Fig. 1. It features eight identical Crank and Slotted-Lever (CSL) 3RP planar mechanisms (where R and P are for revolute and prismatic joints respectively) sharing the same crank and its moving revolute joint, which is centered at point $A(\alpha)$. The common crank is hinged to the base at point O, the eight links with variable length (hereafter called slotted-levers) $A(\alpha)B_i$ (for $i = 1, 2, \ldots, 8$) are hinged to the crank at the common point $A(\alpha)$ and to the base at points B_i, which are symmetrically located with respect to the Y axis along a circular arc having radius $r = OB_i$ and with spread angle 2β (here $\beta = 11.25°$). Possible implementation of the MPR mech-

anism depicted in Fig. 1 could be made by using cylindrical pneumatic or dielectric elastomer 3-way actuators. The output link of the considered MPR is the common crank. A discrete actuation is provided by the eight P joints through identical three-state force generators which, irrespective to the relative position of slider and slotted-lever, supply the forces

$$\mathbf{F}_i = F u_i \left[A\left(\alpha\right) - B_i\right] / \left\|A\left(\alpha\right) - B_i\right\|, \; i = 1, .., 8, \tag{1}$$

with F being a constant magnitude force and u_i being the activation state (namely $u_i \in \{1,\ 0,\ -1\}$) of the i-th actuator. Irrespective to the discrete actuation, the common crank can undergo continuous motion which is limited here in the range $0 \leq \alpha \leq 180°$. By considering all force contributions, the resulting torque M that can be generated at the output crank is

$$M\left(\alpha, u_i\right) = F \sum_{i=1}^{8} u_i \left[\mathbf{k} \cdot \left[A\left(\alpha\right) - O\right] \times \left[A\left(\alpha\right) - B_i\right] / \left\|A\left(\alpha\right) - B_i\right\|\right], \tag{2}$$

where $\boldsymbol{k}$ is the unit vector normal to the plane of motion of the mechanism. Equation (2) represents the static equilibrium condition of the considered ternary MPR (link weight and friction are ignored). For any desired continuous value α^D (i.e. for any desired MPR configuration), the Direct Static Analysis problem amounts to find the torque M^*, within a range $\mathcal{M}$ of discrete values, which corresponds to a known combination of the activation states u_i^D. Conversely, for any α^D, the Inverse Static Analysis (ISA) problem amounts to find the best combination of the activation sates u_i^* (among a total of 3^8 possibilities for each α^D) which enables the generation of the moment M^* (i.e. $M^* = M(\alpha^D,\ u_i^*)$) that more closely matches a desired torque M^D; that is, to find u_i^*, $i = 1, \ldots, 8$, for which the error $e^* = \left\|M^D - M^*\right\|$ is

$$e^* = \min_{u_i \in \{1,0,-1\}} \left(\left\|M^D - M\left(\alpha^D, u_i\right)\right\|\right). \tag{3}$$

Note that since the desired M^D can be any real value, whereas $\mathcal{M}$ is only a discrete subset, in general the minimum error e^* is different than zero. Moreover, owing to the discrete nature of the eight variables u_i, the ISA described by Eq. (3) cannot be solved via standard pseudo-inverse methods. To give an idea of the potential performances of the considered ternary MPR, the ranges $\mathcal{M}$ of available torques that can be generated at the output crank by discretely activating the eight actuators u_i are shown in Figs. 2 and 3 with cyan dot marks (data are computed via Eq. (2) with $F = 10$N, $||A(\alpha) - O\ || = 0.1$m and $r = 0.38$m). In the plots, each line corresponds to

a different angular position of the crank (specifically with α ranging from 0 to 90° and from 15° to 85° with 10° step in Figures 2 and 3 respectively). As shown, despite the discrete activation, this MPR is capable of generating torques in a rather ample range and with a reasonable resolution. Additionally, owing to the possibility of spatially distributing the partitioned actuation system, this MPR also exhibits a rather uniform torque generation capability within its full range of motion ($0 \leq \alpha \leq 180°$). Note that this latter feature cannot be achieved by a standard CSL mechanism actuated by a single continuously regulated force generator.

3 Inverse Static Analysis Models

This section presents five different methods for the solution of the ISA problem described in the previous section, namely: one Look-Up Table model; two Neuro-Fuzzy models; two Neural Network models. Essentially, each of these models is a computational machine that associates an output ternary number $\boldsymbol{u} = [u_1, \ldots, u_8]$ with eight trits (ternary digits) to an input couple of continuum real numbers $\boldsymbol{X} = [X_1, X_2] = [\alpha, M]$. Set-up of all these methods requires the knowledge of an appropriate input-output ($\boldsymbol{X}$-$\boldsymbol{u}$) dataset $\mathcal{D}$ with finite dimensions. Here, $\mathcal{D}$ consists of $10{\cdot}3^8$ $\boldsymbol{X}$-$\boldsymbol{u}$ correspondences that are generated via Eq. (2) for ten different values of α, ranging from 0 to 90° with 10° step, and for all possible (3^8) combinations of $\boldsymbol{u}$ (note that all the $\boldsymbol{X}$-$\boldsymbol{u}$ pairs contained in $\mathcal{D}$ satisfy Eq. (3) with $e^* = 0$). Given the continuity of α, $\mathcal{D}$ is not an exhaustive enumeration of all the possible solutions of the ISA problem. Thus the considered methods are required to provide some generalization ability (namely the ability to find $\boldsymbol{X}$-$\boldsymbol{u}$ pairs for arbitrary α which are not contained within $\mathcal{D}$). To discuss about their suitability for real-time control, the five ISA models are compared in terms of time of off-line preparation t_p, time of on-line calculation t_c, modeling error e_m (i.e. the error calculated via Eq. (3) in predicting $\boldsymbol{X}$-$\boldsymbol{u}$ correspondences for input pairs $\boldsymbol{X}^D = [\alpha^D, M^D]$ contained in $\mathcal{D}$), and generalization error e_g (i.e. the error calculated via Eq. (3) in predicting $\boldsymbol{X}$-$\boldsymbol{u}$ correspondences for input pairs $\boldsymbol{X}^D$ not contained in $\mathcal{D}$).

3.1 Look-Up Table Model

The Look-Up Table (LUT) model is a brute-force solution approach and it is the simplest method considered here. LUT uses a stored data structure as a pattern collection of the entire dataset $\mathcal{D}$ described above. As such, LUT does not require any learning procedure. During model preparation, the input values $\boldsymbol{X}$ of $\mathcal{D}$ are first normalized between 0 and 1, then the so

modified dataset $\mathcal{D}$ is sorted and stored row-by-row in an array. During model usage, the desired inputs $\boldsymbol{X}^D$ are first normalized, second they are compared to the corresponding entries of the LUT using a row-by-row similarity procedure, finally the suitable outputs u_i^* (for $i = 1, .., 8$) are chosen from the LUT row which provides the minimum error between $\boldsymbol{X}^D$ and $\boldsymbol{X}$.

3.2 Neuro-Fuzzy Models

Two Neuro-Fuzzy models are considered which are based on the Neuro-Fuzzy Takagi-Sugeno inference scheme with Gaussian membership functions [7]. Models of this kind are precise fuzzy systems which are static, easy to interpret, focus on accuracy and provide a strong connection between input $\boldsymbol{X}$ and output $\boldsymbol{u}$. Both models are based on the same overall architecture and only differ in the defuzzification operation. In particular, introducing the Gaussian membership functions G_j^n ($j = 1, 2$; $n = 1, \ldots, N$)

$$G_j^n\left(X_j\right) = \exp\left[-\left(X_j - c_j^n/\sigma_j^n\right)^2\right], \tag{4}$$

with characteristic mean c_j^n and variance σ_j^n, together with the fuzzy rules

$$R^n : \text{IF } X_1 \text{ is } G_1^n \text{ AND } X_2 \text{ is } G_2^n \text{ THEN } y_i^n = w_{0i}^n + w_{1i}^n X_1 + w_{2i}^n X_2, \tag{5}$$

with w_{0i}^n, w_{1i}^n and w_{2i}^n being the Takagi-Sugeno weights, the common part of the two Neuro-Fuzzy models calculates the continuous variables

$$\bar{u}_i = \sum_{n=1}^{N} y_i^n \left[\prod_{j=1}^{2} G_j^n\left(X_j\right) \Big/ \sum_{n=1}^{N}\prod_{j=1}^{2} G_j^n\left(X_j\right)\right]. \tag{6}$$

From Eq. (6), the two different models, hereafter briefly referred to as NFTS and NFLUT, are derived by alternatively estimating the actuator activation states u_i through one of the following defuzzification operations

$$u_i = round\left(\bar{u}_i\right) \text{ or } u_i = RLUT\left(\bar{u}_i\right) \tag{7}$$

where $RLUT$ indicates a properly predisposed Reduced Look-Up Table involving $\bar{u}_i$ as only input. Prior to their use, NFTS and NFLUT models require the tuning of the parameters c_j^n, σ_j^n, w_{0i}^n, w_{ji}^n (for $j = 1, 2$; $i = 1, \ldots, 8$; $n = 1, \ldots, N$; in the following $N = 11$). Here, the optimal values of these 308 parameters are found by a learning procedure which employs 12% of the $\boldsymbol{X}$-$\boldsymbol{u}$ pairs known from $\mathcal{D}$. In particular, the learning is performed via the Levenberg-Marquardt Algorithm [8]. Additionally, the NFLUT requires the generation of the RLUT, which is here constructed by storing the most significant $\boldsymbol{u}$-$\overline{\boldsymbol{u}}$ correspondences that occurred during the training with the known dataset $\mathcal{D}$.

3.3 Recurrent Neural Network Models

Two Neural Network models are considered which are based on Elman-type Recurrent Neural Networks with hyperbolic-tangent activation functions. Approximators of this kind are dynamic models that feature short-term memory so as to be capable of representing time-dependent mappings [9]. Both models are based on the same overall architecture and only differ in the presence or absence of the context layer. In particular, for a given input $\boldsymbol{X}(t) = [X_1(t),\, X_2(t)] = [\alpha(t),\, M(t)]$ at the time step t, both models calculate the actuator activation states $u_i(t)$ (for $i = 1, \ldots, 8$) as

$$u_i\,(t) = round\left[G\left(b_{2i} + \sum_{l=1}^{L} w_{il}^{HO} G\,(a_l\,(t))\right)\right] \text{ with } G\,(y) = y\Big/\sqrt{1+y^2}, \tag{8}$$

$$a_l\,(t) = b_{1l} + w^{CL} a_l\,(t-1) + \sum_{j=1}^{2} w_{lj}^{IH} X_j\,(t) \text{ with } a_l\,(0) = 0, \tag{9}$$

where b_{1l}, b_{2i}, w_{il}^{HO}, w_{lj}^{IH} and w^{CL} (for $i = 1, \ldots, 8$; $j = 1, 2$; $l = 1, \ldots, L$; in the following $L = 27$) are model parameters to be determined. From Eq. (8) and (9) the two different models, hereafter briefly referred to as MLP (Multi-Layer Perceptron) and ERNN, are derived by respectively selecting $w^{CL} = 0$ and $w^{CL} = 1/L$. Regarding the remaining 305 parameters, the optimal values are found by a learning procedure which employs 12% of the $\boldsymbol{X}$-$\boldsymbol{u}$ correspondences known from $\mathcal{D}$. In particular, the learning is performed here via an accelerated version of the Back-Propagation Algorithm [10].

3.4 Comparison of the Five Inverse Static Analysis Models

Comparison of the five considered models is reported in Figs. 2 and 3, as well as in Table 1. From these results, it can be concluded:

- LUT provides the worst generalization capabilities and the largest computational time during the on-line phase which makes it unsuited for real-time control applications.

Method/Description	LUT	NFTS	NFLUT	MLP	ERNN
Off-line preparation time t_p (s)	48	965	983	11358	639
On-line computation time t_c (s)	0.3502	0.0014	0.0135	0.0026	0.0026
Modeling error e_m (Nm)	0	0.658	0	0.464	0.377
Generalization error e_g (Nm)	0.805	0.985	0.528	0.515	0.389
Standard deviation of e_g (Nm)	0.499	0.585	0.328	0.431	0.347
Full Scale General. error (%)	16.1	19.7	10.5	10.3	7.8
Note: the CPU has 32 bit operating system, dual core processor, 2.6 GHz, RAM 4 GB.					

Table 1. Performance comparison of the considered methods.

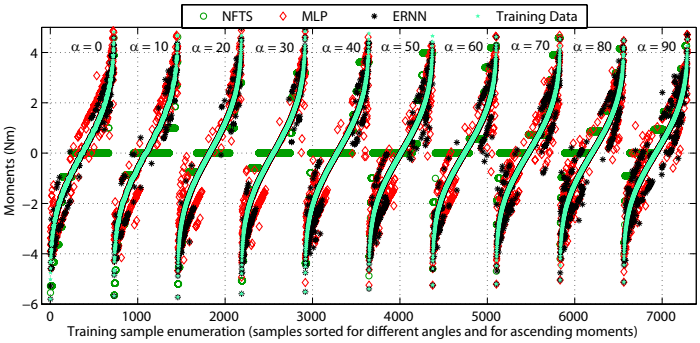


Figure 2. Training performance of different inverse static analysis methods.

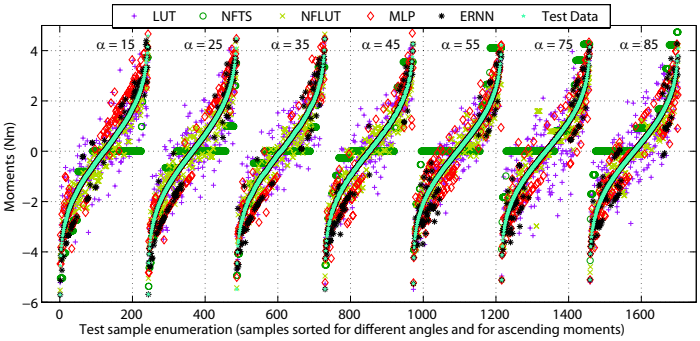


Figure 3. Testing performance of different inverse static analysis methods.

- ERNN is the most accurate model, features the best generalization ability, and requires a rather small computational time during the on-line phase. These features make ERNN very suited for real-time control.
- MLP is comparable to ERNN in terms of modeling accuracy, generalization capability and required on-line computational time. However, it needs very long time for off-line learning.
- NFTS features the shortest on-line computational time; however it is more inaccurate than MLP and ERNN both for modeling and for generalizing.
- NFLUT is rather similar to ERNN in terms of accuracy, but requires a larger on-line computational time.

4 Conclusions

This paper presented: 1) a planar massively parallel robot with 8 three-state force actuators and one continuous degree of rotational motion; 2) one

brute-force method, two Neuro-Fuzzy methods and two Recurrent Neural Network methods for the solution of the inverse static analysis. Thanks to the partitioned and spatially distributed actuator architecture, the considered discrete robot features rather ample, uniform and accurate torque generation capabilities. Comparison among the considered inverse static analysis methods highlighted that Elman type Recurrent Neural Network model is best suited for real-time control applications.

Bibliography

[1] B. Roth, J. Rastegar, and V. Sheinman, "On the design of computer controlled manipulators," in *First CISM-IFTOMM Symposium on Theory and Practice of Robots and Manipulators*, 1973, pp. 93–113.

[2] G. S. Chirikjian, "Inverse kinematics of binary manipulators using a continuum model," *J. of Intelligent and Robotic Systems*, vol. 19, pp. 5–22, 1997.

[3] I. Ebert-Uphoff and G. S. Chirikjian, "Inverse kinematics of discretely actuated hyper-redundant manipulators using workspace densities," in *IEEE Int. Conf. on Robotics and Automation*, 1996, pp. 139–245.

[4] J. Suthakorn and G. S. Chirikjian, "A new inverse kinematic algorithm for binary manipulators with many actuators," *Advanced Robotics*, vol. 15, no. 2, pp. 225–244, 2001.

[5] D. Lichter, V. A. Sujan, and S. Dubowsky, "Computational issues in the planning and kinematics of binary robots," in *IEEE Int. Conf. on Robotics and Automation*, 2000, pp. 341–346.

[6] P. Yang and K. J. Waldron, "Massively parallel actuation," in *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, 2001, pp. 868–873.

[7] J. S. R. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665 –685, 1993.

[8] A. Palit and R. Babuska, "Efficient training algorithm for takagi-sugeno type neuro-fuzzy network," in *IEEE Int. Conf. on Fuzzy Systems*, vol. 3, 2001, pp. 1367–1371.

[9] J. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, pp. 179–211, 1990.

[10] A. K. Palit and D. Popovic, *Computational Intelligence in Time Series Forecasting, Theory and Engineering Applications*. Springer, 2005.

On the addition of degrees of freedom to force-balanced linkages.

V. van der Wijk and Just L. Herder
University of Twente, Faculty of Engineering Technology,
Lab. of Mechanical Automation and Mechatronics, Enschede, The Netherlands.
v.vanderwijk@utwente.nl, j.l.herder@utwente.nl

Abstract The design of shaking-force balanced linkages can be approached by deriving these linkages from balanced linkage architectures. When desired, a possible step is to add degrees-of-freedom (dof), for instance by substituting a link with a n-dof equivalent linkage for which the balanced design of the other links is not affected. This paper shows how the coupler link of a shaking-force balanced 4R four-bar linkage, applied as a 5R five-bar linkage, can be substituted with an equivalent 2-dof pantograph.

1 Introduction

With the increasing speed of manipulators (i.e. mechanisms, robotics), for instance for pick and place tasks, dynamic properties such as shaking-force balance and shaking-moment balance become increasingly important. Acceleration of mass and inertia of moving parts of balanced manipulators do not cause any forces and moments to the base and surrounding, keeping machine vibrations low.

Instead of balancing a manipulator linkage afterwards, it is advantageous to base the design of the linkage on balance properties to minimize complexity, additional mass, and additional inertia (Van der Wijk et al. (2009)). One approach for this is to compose manipulators from balanced linkage sections such as balanced legs (Arakelian and Smith (1999), Wu and Gosselin (2002)). Another approach is to derive manipulators from inherently balanced architectures (Van der Wijk and Herder (2012a)), i.e. linkage architectures that are balanced due to specific kinematic relations. As long as these kinematic relations are maintained, any change to the linkage can be made without affecting the balance properties, for instance by fixing links together and by replacing links with gears.

This paper shows the possibility of adding degrees-of-freedom (dof) to a force-balanced linkage architecture. The substitution of a link with a 2-dof linkage is investigated for which the balanced design of other links is

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*,
CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_7,

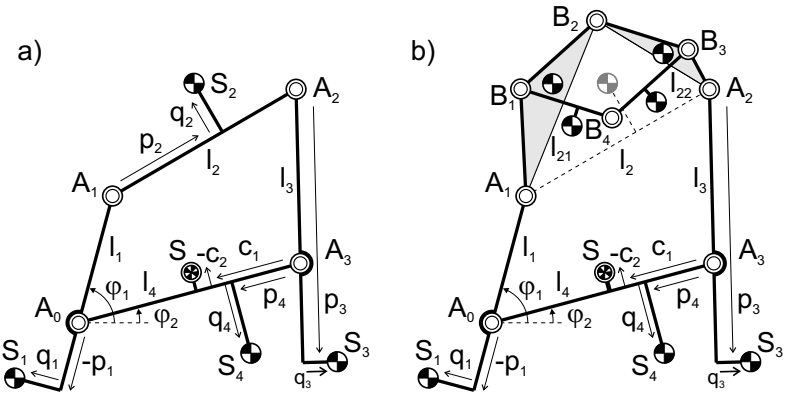


Figure 1. a) Balanced 4R four-bar linkage with CoM at S at link 4; b) Balanced four-bar linkage with coupler l_2 replaced with a pantograph.

not affected. As subject of investigation the coupler link of a shaking-force balanced 4R four-bar linkage, applied as a balanced 5R five-bar linkage, is chosen. First an equivalent model of the coupler link is derived with linear momentum equations and subsequently the conditions for an equivalent 2-dof substitute linkage, a pantograph, are obtained.

2 Equivalent model of coupler link

Figure 1a shows a four-bar linkage $A_0A_1A_2A_3$ of which each link l_i has a mass m_i at its link center-of-mass (link CoM) S_i which are defined with parameters p_i and q_i as indicated. From Berkof and Lowen (1969) and by including the mass m_4 of link 4, the balance conditions for which the CoM of the complete linkage is at an invariant point S in link 4 are written as

$$\begin{array}{lll} p_1 = -\frac{m_2(l_2-p_2)}{l_2}\frac{l_1}{m_1} & q_1 = \frac{m_2 q_2}{l_2}\frac{l_1}{m_1} & c_1 = \frac{1}{m_{tot}}(m_1 l_4 + \frac{m_2(l_2-p_2)}{l_2} l_4 + m_4 p_4) \\ p_3 = l_3 + \frac{m_2 p_2}{l_2}\frac{l_3}{m_3} & q_3 = \frac{m_2 q_2}{l_2}\frac{l_3}{m_3} & c_2 = \frac{1}{m_{tot}}(m_4 q_4 - \frac{m_2 q_2}{l_2} l_4) \end{array} \quad (1)$$

with $m_{tot} = m_1 + m_2 + m_3 + m_4$. The linkage of Fig. 1a then is a force-balanced four-bar linkage when link 4 is stationary with the base and it is a force-balanced five-bar linkage when solely S is stationary with the base as being a movable joint.

To substitute a link without affecting the other links, the substitute linkage has to be equivalent. Therefore first an equivalent model of the coupler link is derived from which the substitute linkage can be found.

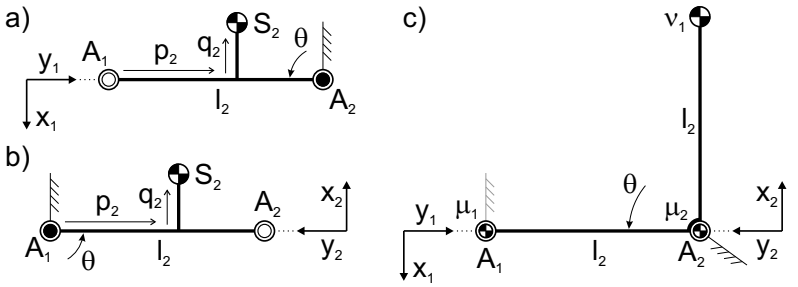


Figure 2. a-b) Coupler link; c) Equivalent Linear Momentum System

With linear momentum equations an equivalent linear momentum system (ELMS) of the coupler can be created. Figures 2a-b show link l_2, of which the linear momentum can be written about A_2 w.r.t. frame x_1y_1 (Fig. 2a) and about A_1 w.r.t. frame x_2y_2 (Fig. 2b) respectively as

$$\frac{\overline{L}_1}{\dot{\theta}} = \begin{bmatrix} m_2(l_2 - p_2) \\ -m_2 q_2 \end{bmatrix} \quad \frac{\overline{L}_2}{\dot{\theta}} = \begin{bmatrix} m_2 p_2 \\ m_2 q_2 \end{bmatrix} \tag{2}$$

Figure 2c shows a model of the coupler with masses μ_1 in A_1, μ_2 in A_2, and mass ν_1 at distance l_2 normal to line A_1A_2 as indicated. Similarly, the linear momentum equations w.r.t. each of the two frames about A_1 and A_2, respectively, can be written as

$$\frac{\overline{L}_1}{\dot{\theta}} = \begin{bmatrix} \mu_1 l_2 \\ -\nu_1 l_2 \end{bmatrix} \quad \frac{\overline{L}_2}{\dot{\theta}} = \begin{bmatrix} \mu_2 l_2 \\ \nu_1 l_2 \end{bmatrix} \tag{3}$$

This implies that the model of Fig. 2c is equivalent to Fig. 2a-b for

$$\mu_1 = \tfrac{m_2(l_2 - p_2)}{l_2} \quad \mu_2 = \tfrac{m_2 p_2}{l_2} \quad \nu_1 = \tfrac{m_2 q_2}{l_2} \tag{4}$$

Also other equivalent models are possible, e.g. with a mass ν_1 at l_2 above A_1. When S_2 is on the line A_1A_2, q_2 is zero for which ν_1 becomes zero too.

3 Equivalent pantograph linkage as substitute

To add one dof, l_2 can be substituted with the 2-dof linkage. Figure 3a shows an ELMS of 2-dof linkage $A_1B_2A_2$ defined with μ_1 in A_1, μ_2 in A_2, and ν_1 at two locations normal to and at equal distance from B_2 as, respectively, l_{21} and l_{22} as indicated. Also here multiple models can be found for equal ELMSs.

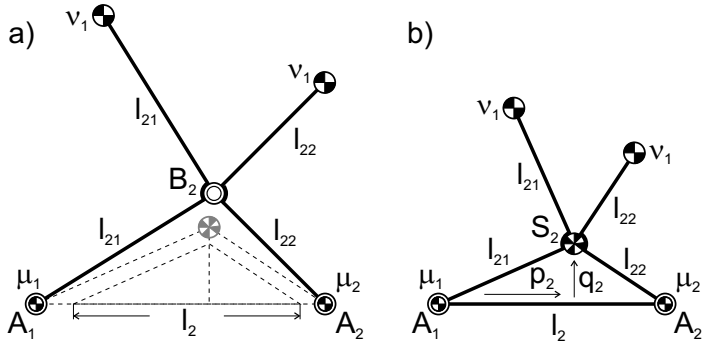


Figure 3. a) Equivalent model of a 2-DoF linkage to replace the coupler; b) Model of the coupler with the CoM of the four masses in S_2.

From Eqs. 1 follows that μ_1, μ_2, and ν_1 need to remain constant to not affect the balance parameters of the other links. Since l_2 is not constant any longer, μ_1, μ_2, and ν_1 can be written as

$$\mu_1 = m_2(1-\kappa_1) \quad \mu_2 = m_2\kappa_1 \quad \nu_1 = m_2\kappa_2 \tag{5}$$

with $\kappa_1 = p_2/l_2$ and $\kappa_2 = q_2/l_2$ to be constant for any value of l_2. This implies that for all lengths l_2, triangle $A_1A_2S_2$ has to be similar. In general a real linkage $A_1B_2A_2$ cannot generate this similarity. A mechanism which is characterized for its properties of similarity is the pantograph linkage (Artobolevskii (1964)). Figure 1b shows how this linkage can be applied to substitute the coupler link.

Figure 4 shows the substitute pantograph linkage in detail, consisting of four links arranged as parallelogram linkage $B_1B_2B_3B_4$ with each a mass m_{2i} located at distances e_{2i} and f_{2i} from joints B_i as indicated. The total mass of the linkage then is written as $m_2 = m_{21} + m_{22} + m_{23} + m_{24}$. The parallelogram linkage, and in specific joints B_1 and B_3, are defined with principal lengths a_1 and a_2 from B_2, respectively, and with angles α_1 and α_2 with the lines A_1B_2 and B_2A_2, respectively.

The pantograph linkage is equivalent to the coupler link when its CoM is located at S_2 of the similar triangle $A_1A_2S_2$ at all times. To find the conditions for which this holds, the linear momentum of the pantograph linkage can be written to be equal to the linear momentum of the ELMS of Fig. 3a. These equations can be written for each dof individually as shown in Van der Wijk and Herder (2012a). The linear momentum for $\dot{\theta}_1$ with

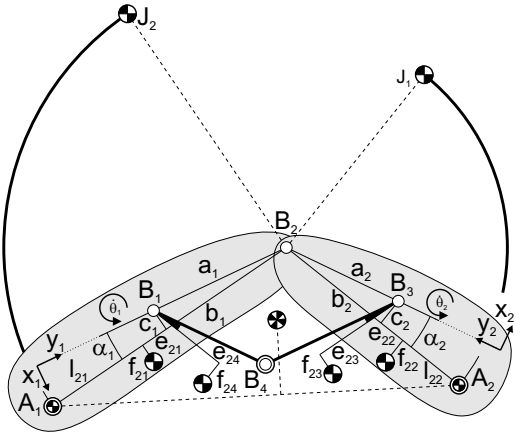


Figure 4. Pantograph linkage $A_1B_1B_2B_3B_4A_2$ and its parameters.

respect to frame x_1y_1 with $\dot{\theta}_2 = 0$ and A_2B_2 fixed can be written as

$$\frac{\overline{L}_{21}}{\dot{\theta}_1} = \begin{bmatrix} u_1 l_{21} \cos\alpha_1 + v_1 l_{21} \sin\alpha_1 \\ u_1 l_{21} \sin\alpha_1 - v_1 l_{21} \cos\alpha_1 \end{bmatrix} = \begin{bmatrix} (m_{21}e_{21} + m_{23}e_{23}) \cos\alpha_1 - (m_{21}f_{21} + m_{23}f_{23}) \sin\alpha_1 + m_{24}a_1 \\ (m_{21}e_{21} + m_{23}e_{23}) \sin\alpha_1 + (m_{21}f_{21} + m_{23}f_{23}) \cos\alpha_1 \end{bmatrix} \tag{6}$$

and the linear momentum for $\dot{\theta}_2$ with respect to frame x_2y_2 with $\dot{\theta}_1 = 0$ and A_1B_2 fixed writes

$$\frac{\overline{L}_{22}}{\dot{\theta}_2} = \begin{bmatrix} u_2 l_{22} \cos\alpha_2 + v_1 l_{22} \sin\alpha_2 \\ -u_2 l_{22} \sin\alpha_2 + v_1 l_{22} \cos\alpha_2 \end{bmatrix} = \begin{bmatrix} (m_{22}e_{22} + m_{24}e_{24}) \cos\alpha_2 - (m_{22}f_{22} + m_{24}f_{24}) \sin\alpha_2 + m_{23}a_2 \\ -(m_{22}e_{22} + m_{24}e_{24}) \sin\alpha_2 - (m_{22}f_{22} + m_{24}f_{24}) \cos\alpha_2 \end{bmatrix} \tag{7}$$

These equations lead to the resulting four conditions for equivalence

$$\begin{array}{llll} X_{11}\cos\alpha_1 + X_{12}\sin\alpha_1 = & m_{24}a_1 & X_{21}\cos\alpha_2 + X_{22}\sin\alpha_2 = & m_{23}a_2 \\ X_{11}\sin\alpha_1 - X_{12}\cos\alpha_1 = & 0 & X_{21}\sin\alpha_2 - X_{22}\cos\alpha_2 = & 0 \end{array} \tag{8}$$

with

$$\begin{array}{ll} X_{11} = u_1 l_{21} - m_{21}e_{21} - m_{23}e_{23} & X_{12} = v_1 l_{21} + m_{21}f_{21} + m_{23}f_{23} \\ X_{21} = u_2 l_{22} - m_{22}e_{22} - m_{24}e_{24} & X_{22} = v_2 l_{22} + m_{22}f_{22} + m_{24}f_{24} \end{array} \tag{9}$$

When the mass of each link and the link-CoMs are known, the locations of joints B_1 and B_3 are found with

$$\begin{aligned} \tan\alpha_1 &= \frac{X_{12}}{X_{11}} \qquad a_1 = \frac{1}{m_{24}}(X_{11}\cos\alpha_1 + X_{12}\sin\alpha_1) \\ \tan\alpha_2 &= \frac{X_{22}}{X_{21}} \qquad a_2 = \frac{1}{m_{23}}(X_{21}\cos\alpha_2 + X_{22}\sin\alpha_2) \end{aligned}$$

or by substituting

$$\begin{array}{lll} \cos\alpha_1 = \frac{b_1}{a_1} & \sin\alpha_1 = \frac{c_1}{a_1} & a_1^2 = b_1^2 + c_1^2 \\ \cos\alpha_2 = \frac{b_2}{a_2} & \sin\alpha_2 = \frac{c_2}{a_2} & a_2^2 = b_2^2 + c_2^2 \end{array}$$

in the four conditions of Eqs. 8, which results in

$$\begin{array}{llll} X_{11}b_1 + X_{12}c_1 = & m_{24}(b_1^2 + c_1^2) & X_{21}b_2 + X_{22}c_2 = & m_{23}(b_2^2 + c_2^2) \\ X_{11}c_1 - X_{12}b_1 = & 0 & X_{21}c_2 - X_{22}b_2 = & 0 \end{array} \tag{10}$$

algebraic solutions are obtained for a_1, b_1, and c_1 being

$$a_1 = \frac{1}{m_{24}}\sqrt{X_{11}^2 + X_{12}^2}, \quad b_1 = \frac{X_{11}}{m_{24}}, \quad c_1 = \frac{X_{12}}{m_{24}} \tag{11}$$

and for a_2, b_2, and c_2 being

$$a_2 = \frac{1}{m_{23}}\sqrt{X_{21}^2 + X_{22}^2}, \quad b_2 = \frac{X_{21}}{m_{23}}, \quad c_2 = \frac{X_{22}}{m_{23}} \tag{12}$$

A pantograph linkage according the conditions of Eqs. 8 can replace link l_2 of Fig. 1a as shown in Fig. 1b without affecting the other links for perfect force balance of the complete linkage for all motion.

4 Discussion

In addition to substituting the coupler link, also any of the other three links can be substituted with equivalent linkages to add dofs. Figure 5a shows the result when each of the four links is substituted with an equivalent pantograph with which the mechanism gains four dofs. The procedure to derive the conditions for equivalence is similar to the procedure for the coupler link with some differences due to their specific position within the linkage. Unfortunately this article leaves too few space to discuss them here in detail.

Both branches of the resulting equivalent pantographs can be used, the choice does not affect the design parameters. From Fig. 5a it is observed

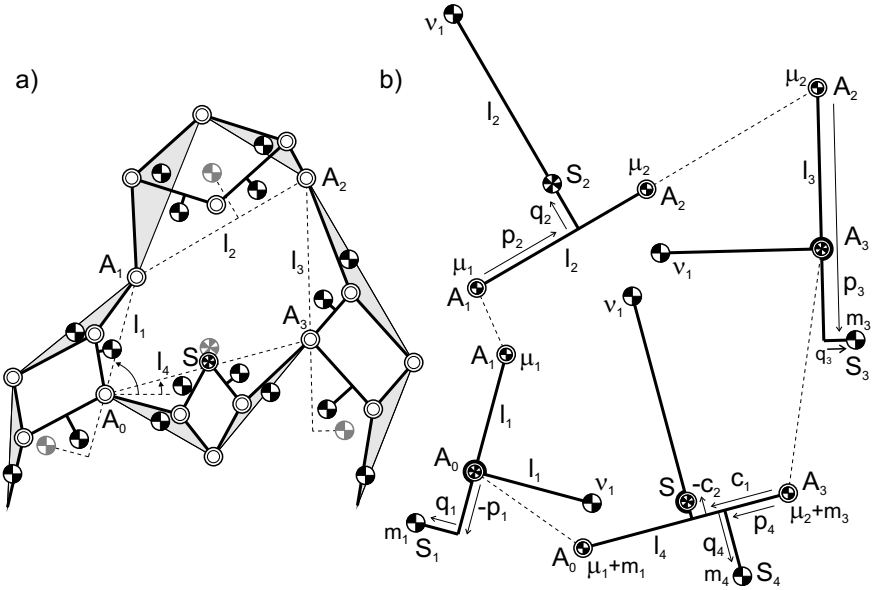


Figure 5. a) Resulting linkage when all four links are substituted with a pantograph gaining four dofs; b) With μ_i and ν_1 the mass of the coupler can be distributed equivalently to the other links. For balancing the linkage then each of the three other links can be separately investigated. For links 1 and 3 the CoM of its link mass and the equivalent masses is in joints A_0 and A_3, respectively. The linkage CoM S at link 4 is the CoM of its equivalent system.

that the coupler substitute pantograph is in the other branch as compared to the other three pantographs.

With the approach in this article it is also possible to substitute links with equivalent 3-dof or higher-dof linkages. These linkages will be pantographic linkages as investigated in Van der Wijk and Herder (2012a), consisting of multiple parallelograms.

The equivalent masses μ_i and ν_1 share another feature which is illustrated in Fig. 5b and which can be obtained from Eqs. 1. With μ_i and ν_1 modeled at link 2 as indicated, S_2 is characterized as the CoM of the three equivalent masses. With μ_1 and ν_1 modeled at link 1 as indicated, A_0 is the CoM of the two equivalent masses and mass m_1. With μ_2 and ν_1 modeled at link 3 as indicated, A_3 is the CoM of the two equivalent masses and mass m_3. With the three equivalent masses modeled at link 4 as indicated and with m_1 in A_0 and m_3 in A_3, S is the CoM of the complete model. μ_i and

ν_1 therefore can be interpreted as the distribution of the mass of link 2 onto the other three links. When linkages become complex, this feature is useful for finding the balance conditions of linkages with arbitrary mass distributions without the need of loop equations, which was shown in Van der Wijk and Herder (2012b). There can be various equivalent models. For instance the initial coupler link can also be modeled as in Fig. 3b for which S_2 is the CoM of the four equivalent masses.

5 Conclusion

For the purpose of adding degrees of freedom (dof) to shaking-force balanced linkages, the coupler link of a 4R four-bar linkage was substituted with a 2-dof equivalent pantograph linkage. An equivalent model of the coupler link was derived with linear momentum equations with which the conditions for the pantograph linkage were determined. It was discussed that the other links can be substituted in a similar way and that also higher-dof equivalent linkages can be used as substitutes. In addition, it was shown how equivalent masses can be used to distribute the coupler mass to the other three links.

Bibliography

V. G. Arakelian and M. R. Smith. Complete shaking force and shaking moment balancing of linkages. *Mechanism and Machine Theory*, 34: 1141–1153, 1999.

I. I. Artobolevskii. *Mechanisms for the Generation of Plane Curves*. Pergamon Press, 1964.

R. S. Berkof and G. G. Lowen. A new method for completely force balancing simple linkages. *J. of Engineering for Industry*, pages 21–26, February 1969.

V. Van der Wijk and J. L. Herder. Synthesis method for linkages with center of mass at invariant link point - pantograph based mechanisms. *Mechanism and Machine Theory*, 48:15–28, 2012a.

V. Van der Wijk and J. L. Herder. Inherently balanced 4r four-bar based linkages. *Proc. of the 13th International Symposium on Advances in Robot Kinematics (ARK), Innsbruck, Austria (accepted)*, 2012b.

V. Van der Wijk, J. L. Herder, and B. Demeulenaere. Comparison of various dynamic balancing principles regarding additional mass and additional inertia. *Journal of Mechanisms and Robotics*, 1(4):04 1006, 2009.

Y. Wu and C. M. Gosselin. On the synthesis of reactionless spatial 3-dof mechanisms using planar four-bar linkages. *Proc. of ASME DETC, Montreal, CA.*, 2002. DETC2002/MECH-34311.

Chapter II

Parallel Robots Design, Calibration and Control

Design Challenges in the Development of Fast Pick-and-place Robots

Jorge Angeles*

* Department of Mechanical Engineering & Centre for Intelligent Machines, McGill University, Montreal, QC, Canada

Abstract The development of robotic systems has faced many challenges: First came what Freudenstein called "the Mount Everest" of kinematics. Thereafter came the challenge of finding all forward-kinematics solutions of a six-degree-of-freedom parallel robot. The two foregoing problems are largely solved by now. The new challenge is the development of ever faster pick-and-place four-degree-of-freedom robots. The limit of the serial version thereof was reached in the late nineties, with a record speed of two cycles per second. This called for the development of parallel versions of the same. Some industrial robots of this kind, carrying three to four limbs, are out in the market. With the purpose of simplifying their morphology and reducing their footprint, two-limb robots have started emerging. The challenge here is the transmission of force and motion from the two actuators of each limb, mounted on a common base, to produce two independent motions, normally pan and tilt. Discussed in this paper are the theoretical and practical hurdles that the robot designer faces in this quest.

1 Introduction

The early robots, namely, *Unimate* (Westerlund, 2000) and the Stanford Arm, had a simple architecture, intended to ease the transformation from Cartesian to joint coordinates and vice versa, which led their designers to provide them with one prismatic joint. The need for robots with a more dextrous architecture, which required six revolute joints, brought about the challenge of the *inverse kinematics* of a serial robot supplied with six revolute joints, but otherwise of arbitrary architecture. This problem is equivalent to that of finding the input-output function of the closed seven-revolute linkage, which Freudenstein dubbed "the Mount Everest of kinematics." Using different elimination procedures, Li (1990), Li et al. (1991) and Raghavan and Roth (1990), devised independent algorithms for the computation

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_8,

of the coefficients of the 16th-degree univariate polynomial that had been anticipated by Primrose (1986).

Once Freudentstein's Mount Everest problem was under control, researchers turned their attention to the next challenge, namely, the derivation of the minimal univariate polynomial that would yield all the *forward kinematics* solutions of the general Stewart-Gough platform (SGP). This is a six-degree-of freedom parallel robot with two platforms coupled by six limbs, of arbitrary architecture. Independently, Wampler (1996) and Husty (1996) devised procedures to derive this polynomial, although Wampler did not pursue the univariate polynomial approach and preferred to cast the problem in a form suitable for its solution by means of polynomial continuation. Husty did derive the 40th-degree univariate polynomial.

The two foregoing challenges closed an important chapter, that had kept researchers busy for a good 30 years, in the development of robots with serial and parallel architectures. The algebraic-geometric complexity of these two major robotics problems led kinematics researchers to overlook what appeared kinematically a light problem: the kinematics of SCARA (Selective Compliance Assembly Robot Arm) systems, that had appeared in the early eighties. These robots exhibit a quasi-planar architecture, with three revolute (R) and one prismatic (P) joints, the former of parallel axes, thereby forming a planar three-degree-of-freedom (dof) sub-manipulator. The P joint allows for pure rigid-body translations in a direction normal to the plane of the sub-manipulator. This novel architecture was introduced in 1981 by Prof. Hiroshi Makino, of Yamashi University, in Japan (Westerlund, 2000). SCARA systems targeted fast pick-and-place operations (PPO), as those found in the assembly of electronic devices with a flat geometry.

SCARA robots were fully developed by the end of the 20th century, with Adept Technology boasting cycle times of 2 Hz by their COBRA s600 robot. The industry-adopted cycle consists of the path displayed in Fig. 1: a landmark point of the robot end-effector (EE), termed the operation point or, more popularly, the tool-center point, must traverse this trajectory from point 1 to point 4, and return to point 1; while on the 300-mm segment, from point 2 to point 3, the EE, moreover, must turn 180° ccw when viewed from the positive direction of the z-axis, 180° cw upon returning.

However, the market was asking for faster robots and for robots with a higher load-carrying capacity. ABB Robotics came up with a novel architecture, the ABB IRB series, that features a kinematic chain that carries a Π joint, namely, a planar parallelogram linkage. This joint is rather new in the art, as it didn't appear as a kinematic joint but only in the early nineties. Wohlhart (1991), Hervé and Sparacino (1992) and Dietmaier (1992) independently reported work on the analysis of linkages with Π joints in their

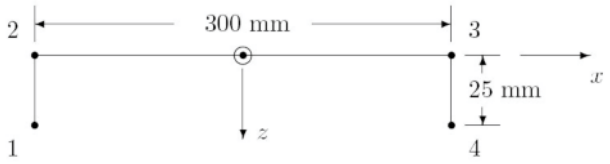


Figure 1. The industry-adopted test cycle for SCARA systems

chain—the name of the joint had been proposed by Hervé and Sparacino. ABB Robotics produced a four-dof family of robots with a larger load-carrying capacity than that of the well-known SCARA systems, by means of a kinematic chain of the RΠΠR type. Although not billed as a SCARA system, this family is in fact so. However, this robot is not intended for fast PPO.

In the late nineties, it became apparent that, in terms of speed, serial SCARA systems had reached their limits, as their architecture transfers the static and inertial load of their upstream motors to their downstream counterparts. This realization led to the emergence of parallel SCARA systems.

2 The Development of Parallel SCARA Systems

A robot structure that appeared as a suitable platform to develop a parallel SCARA was Clavel's Delta (Clavel, 1988). This is how ABB Robotics and Fanuc Robotics developed their FlexPicker and M1*i*A robots, respectively. The two robots have one similar architecture: a parallel substructure, the Delta layout, to produce three independent translations of its moving platform (MP). A fourth motion, rotation about a vertical axis, is produced by means of a fourth motor fixed to the base platform (BP), which transmits its motion to the gripper—articulated via a revolute joint to the MP—by means of a telescopic Cardan shaft. The latter comprises a kinematic chain of the UPU type, where U stands for *universal joint*. This layout makes of these two robots *hybrid*, or serial-parallel.

A fully parallel architecture of a robot capable of the motions of SCARA systems, a.k.a. Schönflies-motion generator (SMG), was patented by a French-Japanese team, who dubbed it the *H4 Robot* (Company et al., 2001). This robot features four identical limbs coupling a base platform with a moving platform. Each limb carries four R joints and one modified Π joint. The latter is a regular Π joint with its R joints replaced with orientable pin joints, which amount to spherical joints. The purpose of the modification is

to allow for the assemblability of the robot. Moreover, the limbs are driven by identical direct-drive motors, thereby doing away with gears and their well-known drawbacks, mainly backlash and dry friction. Furthermore, the four limbs are arrayed in two consecutive pairs, each carrying an end-plate. The two end plates are then used to drive the gripper by means of a gear mechanism. The purpose of this mechanism is to allow for a rotation of 180° of the gripper.

The complexity of the gripper-turning mechanism and of the large number of limbs, which entail the danger of limb collision, has prompted the development of simpler architectures.

3 Two-limb Schönflies-motion Generators

One two-limb SMG was proposed by Angeles et al. (2006) to overcome the hurdles of four-limb SMGs. Lee (1990), in turn, proposed four two-limb architectures for the same purpose, one of these displayed in Fig. 2. A major difference between the latter and the former is that the latter are *isoconstrained*—i.e., isostatic—structures, the former *overcostrained*. An advantage of isoconstrained kinematic chains is that they can be assembled in spite of the unavoidable manufacturing errors, without introducing residual stresses. The two limbs of all of Lee's instances bear identical topologies, with one minor geometric difference, as described below: the gripper is rigidly attached to the MP, and coupled to the base platform by means of limbs with a kinematic chain of the CRRH type, where C stands for cylindrical and H for screw joint, the former allowing for two-dof relative motion between the coupled links; a condition for the functioning of this robot is that the leads of the two end H joints be distinct. Application of the *Chebyshev-Grübler-Kutzbach formula* (Angeles, 2005) then yields four dof. That is, the chain is isoconstrained. Moreover, a group-theoretical analysis of the linkage, as outlined in the foregoing reference, can be applied to show that the only common motion subset of the two limb-kinematic bonds—an extension of the concept of kinematic pair (Hervé, 1999), applied to open kinematic chains comprising several kinematic pairs—is the *Schönflies subgroup*. Therefore, the closed chain is a SMG.

Now, since a SMG is endowed with four dof, it requires four independent actuators that might as well be mounted on the BP. Two-limb SMGs thus require two motors per limb mounted onto the BP. Moreover, if *complexity* is introduced as a design criterion at the conceptual stage of the design process (Khan and Angeles, 2011), then the two motors driving each limb might as well share equally the total load, which requires novel, suitable drives. One more design challenge is the stiffness of the structure, as two

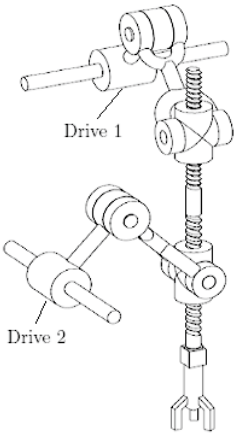


Figure 2. A two-limb isoconstrained SMG

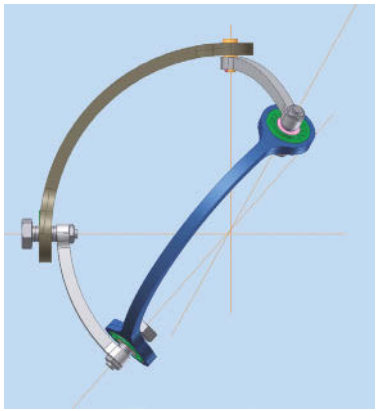


Figure 3. An instance of a spherical homokinetic mechanism

limbs, in principle, provide a lower stiffness than four. Finally, a mechanism is required to *remotely* rotate the gripper through a 180° turn in both directions. These items are discussed below.

Design Challenge 1: The Drives The McGill SMG features two driving Π joints, one per limb. The links of each Π joint, moreover, are made of carbon fiber, and mounted on a common base that can pan to properly orient the vertical planes of the parallelograms. Additionally, each drive provides a tilt motion to each joint, which is used to translate the MP in a given plane. The two-dof drive unit was designed based on a planetary gear train whose sun and ring gears are driven by identical motors. The pan is obtained directly from the rotation of the planet-carrier, the tilt from that of the planets about their own axes via a bevel gear train that transforms the rotation of the planets about their vertical axes into a rotation about a horizontal axis (Angeles et al., 2006). The problem with this drive is the multitude of gears, which bring about serious backlash problems. Details of this drive are recorded in (Cammarata et al., 2010).

Alternatives to the foregoing drives have been devised to produce the tilt motion, namely, *i*) a five-bar linkage of the RHRRR type (Alizadeh et al., 2012) and *ii*) a *quasi-homokinetic* spherical linkage (Alizadeh et al., 2011). The sobriquet of the latter aims to hint that the spherical linkage produces a homokinetic transmission both approximately and only in a prescribed

range, of 120°, of the input rotation. An instance of this linkage is illustrated in Fig. 3, which features a robust design in that it accommodates manufacturing errors by means of a concatenation of a R and a H kinematic pairs at the passive joints. Such a concatenation produces a C joint, according to the theory of displacement groups. As argued in the above reference, the RH realization of the C joint is preferred to its better-known RP counterpart because of the *robustness* of the former.

The C joints of the SMG of Fig. 2 require a slide-roll drive, i.e., a mechanism that is capable of driving the two dof of the C joint. Again, upon resorting to the theory of displacement subgroups, it is possible to find solutions that otherwise would be elusive. Such a mechanism is the cylindrical differential linkage[1] of Fig. 4. The collar of this linkage is the nut of two independent screws of identical leads, with one left-, one right-handed.

Figure 4. A RHHR linkage designed to drive a C joint

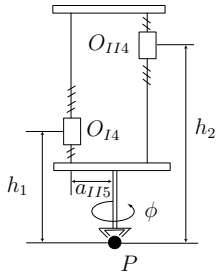


Figure 5. A gripper-turning double-screw linkage

Design Challenge 2: A Stiff Structure In order to enhance the stiffness of the structure of parallel SMGs, links made of composite materials, mainly carbon fiber, are often used. The two longer links of the Π joints of the McGill SMG are fabricated as boxes of carbon fiber. However, in two-limb structures, this is not sufficient. The McGill SMG entails a symmetric structure of the RΠΠΠRRΠΠΠR type, with a home posture in which the four Π joints lie in the same plane. This renders the structure highly flexible in a direction normal to that plane. The structure of the robot of Fig. 2 appears

[1]The author assigned the problem of devising *the simplest* drive for the C joint to his senior UG/graduate students of a course on kinematic synthesis; four out of 12 were capable of synthesizing the foregoing RHHR linkage.

much stiffer, by virtue of the relative layout of the axes of its two C joints, namely, offset and at right angles. In this robot, the two proximal links of each limb can be made of carbon fiber.

Design Challenge 3: The Gripper-turning Mechanism The turning mechanism of the H4 SMG appears to be highly complex, with a multitude of gears—reportedly, the gears have been replaced by belts and pulleys. On the other hand, the double-screw mechanism proposed by Lee, as shown in Fig. 2, is extremely simple, and hence, suitable for the turning operation. An alternative double-screw mechanism, proposed for the McGill SMG, is shown in Fig. 5. It consists of two parallel screws of identical leads but of opposite hands, and parallel axes (Alizadeh, 2012).

4 Conclusions

The challenges faced by the designers of fast pick-and-place robots of the Schönflies type were outlined here, with a critical assessment of their pros and cons. A novel drive for C joints was introduced, whose realization was made possible by the theory of displacement subgroups.

Bibliography

D. Alizadeh. *Optimum Kinetostatic Design of Parallel Two-limb Schönflies-motion Generators–Ph.D. Thesis.* McGill University, 2012.

D. Alizadeh, J. Angeles, and S. Nokleby. Optimization of a spherical 'homokinetic' linkage with minimum design error and maximum transmission quality. In *Proceedings of the 13th IFToMM World Congress*, pages Paper A–354, 2011.

D. Alizadeh, X.Q. Ma, and J. Angeles. On the reduction of the normality conditions in equality-constrained optimization problems in mechanics. *Meccanica*, pages 755–768, 2012.

J. Angeles. The degree of freedom of parallel robots: a group-theoretic approach. In *Proceedings of the IEEE Int. Conf. Robotics & Automation*, pages 1017–1024, 2005.

J. Angeles, S. Caro, W. Khan, and A. Morozov. Kinetostatic design of an innovative Schönflies-motion generator. *Proc. IMechE Part C: J. Mechanical Engineering Science*, 220:935–943, 2006.

A. Cammarata, J. Angeles, and R. Sinatra. Kinetostatic and inertial conditioning of the McGill Schönflies-motion generator. *Advances in Mechanical Engineering*, page DOI: 10.1155/2010/186203, 2010.

R. Clavel. Delta, a fast robot with parallel geometry. In *Proceedings of the 18th Int. Symp. Industrial Robots*, pages 91–100, 1988.

O.A. Company, F. Pierrot, T. Shibukawa, and K. Morita. *Four-degree-of-freedom parallel robot*, Patent EP1084802 edition, 2001.

P. Dietmaier. Inverse kinematics of manipulators with 3 revolute and 3 parallelogram joints. In *Proceedings ASME 22nd. Biennial Mechanisms Conference*, pages 35–40, 1992.

J.M. Hervé. The Lie group of rigid body displacements, a fundamental tool for mechanism design. *Mechanism and Machine Theory*, pages 719–730, 1999.

J.M. Hervé and F. Sparacino. Star, a new concept in robotics. In *Proc. 3rd Int. Workshop on Advances in Robot Kinematics*, pages 176–83, 1992.

M.L. Husty. An algorithm for solving the direct kinematics of general Stewart-Gough platforms. *Mechanism and Machine Theory*, 31:365–380, 1996.

W.A. Khan and J. Angeles. A novel paradigm for the qualitative synthesis of simple kinematic chains based on complexity measures. *ASME Journal of Mechanisms and Robotics*, pages 031010–1–031010–11, 2011.

C.C. Lee. Isoconstrained mechanisms for fast pick-and-place manipulation. In *Proceedings 1st International Symposium on Geometric Methods in Robotics and Mechanism Research*, pages 95–112, 1990.

H. Li. Ein Verfahren zur vollständigen Lösung der Rückwärtstransformation für Industrieroboter mit allgemeiner Geometrie, Doctoral Thesis. Universität-Gesamthochschule Duisburg, 1990.

H. Li, C. Woernle, and M. Hiller. A complete solution for the inverse kinematic problem of the general 6R robot manipulator. *ASME J. Mechanical Design*, 113:481–486, 1991.

E.J.F. Primrose. On the input-output equation of the general 7R mechanism. *Mechanism and Machine Theory*, 21:509–510, 1986.

M. Raghavan and B. Roth. Kinematic analysis of the 6R manipulator of general geometry. In *Proc. Fifth Int. Symposium on Robotics Research*, pages 263–270, 1990.

C.W. Wampler. Forward displacement analysis of general six-in-parallel SPS (Stewart) platform manipulators using soma coordinates. *Mechanism and Machine Theory*, 31:331–337, 1996.

L. Westerlund. *The Extended Arm of Man. A History of the Industrial Robot.* Informationsförlaget, 2000.

K. Wohlhart. Der homogene Paralleltrieb-Mechanismus. *Mathematica Pannonica*, 2:59–76, 1991.

A Low Energy Consumption Solar Tracker based in Parallel Kinematics

Oscar Altuzarra[‡], Iván Seras[†], Erik Macho[‡], Jokin Aginaga[‡*]

[‡] Department of Mechanical Engineering, University of the Basque Country, Bilbao, Spain

[†] EDAI, Vizcaya, Spain

Abstract Solar tracking devices try to increase the area exposed to direct radiation of the sun. The main drawback of these kind of devices is the energy consumed when following the sun. This work presents the design of a mechanism able to accurately follow the sun motion minimizing the energy consumption during its operation. The objective is achieved by means of a mechanism with parallel kinematics architecture.

1 Introduction

Increasing the efficiency of photovoltaic collectors requires efforts in various directions. One of them is the design of an efficient solar tracker. The aim of this kind of devices is to increase the area exposed to direct radiation of the sun. Since the diurnal and seasonal movement of earth affects the radiation intensity on the solar collectors, solar trackers move the solar systems to compensate for these motions, keeping the collector perpendicular to the sun. Although solar trackers improve the amount of energy absorbed, maintaining the collectors perpendicular to the sun radiation involves an energy consumption. Then, a design requeriment is to minimize the energy consumed.

A wide review of solar tracking systems is presented in Mousazadeh et al. (2009). Solar trackers are divided into passive and active trackers, and the latter in turn are divided in sensorized ones and programmed ones. Another division can be made taking into account the mechanical characteristics. From such point of view, solar trackers are usually divided in those with

[*]The authors of this paper wish to acknowledge the finance received from the Spanish Government via the Ministerio de Educación y Ciencia (Projects DPI2008-00159 and DPI2011-22955), the ERDF of the European Union and the Government of the Basque Country (Project GIC07/78, IT445-10).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_9,

a unique axis and those with two axes. The former are divided in vertical axis trackers, as in Lorenzo et al. (2002), horizontal axis trackers, as in Chin et al. (2011), and inclinated axis trackers. Solar trackers with one axis have much better performance than fixed systems, but two-axis systems allow to obtain an optimal tracking of the sun's path, since they keep the orientation of the collectors perpendicular to the solar radiation at any time in any season. The most popular tracking systems with one or two axes are shown in Figure 1.

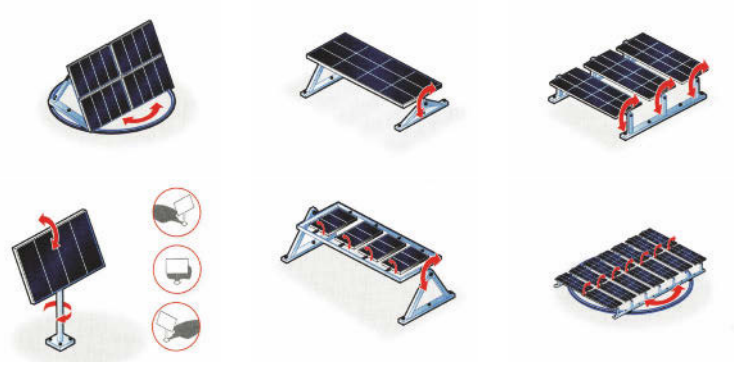


Figure 1. Popular architectures for solar tracking

This work proposes a new solar tracking system which moves the collector exploiting gravitational potential energy. This idea is carried out by means of a parallel kinematics mechanism (Merlet (2006)), in which the collector is attached to the mobile platform. The collector is raised to certain height at night and from the sunrise it is slowly dropped keeping the desired orientation by controlling the brakes of the actuators. This way, the energy consumption of the solar tracker is minimized.

2 Mechanism Description and Kinematic Analysis

The mechanism developed for this application is the four degrees-of-freedom parallel platform depicted in Figure 2.

This platform has two identical $\underline{P}RRR$ kinematic chains, $i = 1, 2$, and and other two $\underline{P}RRS$ kinematic chains. In the first $\underline{P}RRR$ chain, the actuated prismatic joint is vertical, the first revolute joint is horizontal **j** and joined to the slider B_1, the second revolute joint is parallel to the first one and passes by the end C_1 of the rigid bar B_1C_1, and the last revolute joint is located at C_1 and oriented toward the center P of the end-effector

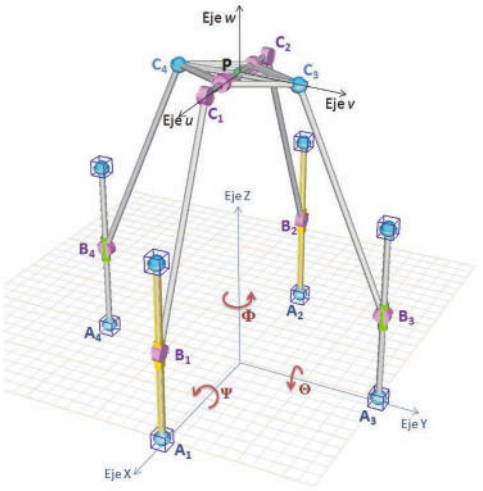


Figure 2. Kinematic model of the four degrees-of-freedom platform

$(\mathbf{c}_i - \mathbf{p})$. It can be noticed that the mechanism made up by both chains and the middle line of the moving platform is a planar mechanism on the XZ plane. This means that the reference point P has a fixed coordinate $Y = 0$ which avoids the movement of the moving platform along the direction of such an axis. In addition to this, revolute joints avoid any rotation of the planar moving platform around the direction of its perpendicular $\mathbf{j} \times (\mathbf{c}_i - \mathbf{p})$, which is defined by the axes of the final chains' revolute joints. Thus, this constraint analysis make evident the fact that the moving platform has four degrees-of freedom, two translational and two rotational ones.

The output motion type can be formally explained as follows. The platform element twist $\$_p$ resulting form the kinematic chain 1 can be obtained as a linear combination of four linearly independent screws associated with each kinematic joint of such a chain, thus forming a 4-system. These screws are: $\hat{\$}_{1,1}$ associated with the prismatic joint of point B_1, step $h = \infty$; $\hat{\$}_{1,2}$ associated with the revolute joint of point B_1, step $h = 0$; $\hat{\$}_{1,3}$ associated with the revolute joint of point C_1, step $h = 0$; and $\hat{\$}_{1,4}$ associated with the second revolute joint of point C_1, step $h = 0$.

$$\hat{\$}_{1,1} = \begin{bmatrix} \mathbf{0} \\ \mathbf{k} \end{bmatrix} \quad \hat{\$}_{1,2} = \begin{bmatrix} \mathbf{j} \\ \mathbf{b}_1 \times \mathbf{j} \end{bmatrix} \quad \hat{\$}_{1,3} = \begin{bmatrix} \mathbf{j} \\ \mathbf{c}_1 \times \mathbf{j} \end{bmatrix} \quad \hat{\$}_{1,4} = \begin{bmatrix} \mathbf{v} \\ \mathbf{c}_1 \times \mathbf{v} \end{bmatrix} \tag{1}$$

being $\mathbf{j}$, $\mathbf{k}$ unit vectors along the fixed axes directions Y and Z, $\mathbf{b}_1$ and $\mathbf{c}_1$

position vectors of points B_1 y C_1 from P, and $\mathbf{v}$ the direction of the last revolute joint, which coincides with line C_1P.

The kinematic screw resulting in the end-effector is obtained from:

$$\$_P = \sum_{j=1}^{4} \dot{q}_{1j} \hat{\$}_{1,\mathrm{j}} \tag{2}$$

where $\dot{q}_{1j}$ is the velocity intensity at joints j of chain 1.

It is possible to obtain principal screws forming a generalized basis of this motion by means of mathematical tools well established in this theory. In this case, the result is a basis made up of:

$$\hat{\$}_{\mathrm{p},1} = \begin{bmatrix} \mathbf{0} \\ \mathbf{i} \end{bmatrix} \qquad \hat{\$}_{\mathrm{p},2} = \begin{bmatrix} \mathbf{0} \\ \mathbf{k} \end{bmatrix} \qquad \hat{\$}_{\mathrm{p},3} = \begin{bmatrix} \mathbf{j} \\ \mathbf{c}_1 \times \mathbf{j} \end{bmatrix} \qquad \hat{\$}_{\mathrm{p},4} = \begin{bmatrix} \mathbf{v} \\ \mathbf{0} \end{bmatrix} \tag{3}$$

so the screws of this kinematic chains constitute a 4-system of 2-$\hat{\$}_\infty$ and 2-$\hat{\$}_0$ type.

In order to analyze the motion constraints due to kinematic joints it is necessary to make a study of the manipulator's end-effector *wrench* (set of forces and torques which limit its motion). Since this is complementary to the kinematic screws system has dimension 2 and type 1-$\hat{\$}_0$ and 1-$\hat{\$}_\infty$. This wrench is defined by 2 *reciprocal screws* to the moving platform twist. These screws are:

$$\hat{\$}_{\mathrm{r},1} = \begin{bmatrix} \mathbf{j} \\ \mathbf{0} \end{bmatrix} \qquad\qquad \hat{\$}_{\mathrm{r},2} = \begin{bmatrix} \mathbf{0} \\ \mathbf{j} \times \mathbf{v} \end{bmatrix} \tag{4}$$

The first screw of the moving platform wrench has step $h = 0$ and represents a force in the Y axis direction applied in P, while the second screw has step $h = \infty$ and represents a torque around the direction perpendicular to the revolute joint axes ($\mathbf{j}$ and $\mathbf{v}$). Therefore, this means the only moving platform motions avoided are the translation along the direction of Y axis and the rotation around the direction defined by $\mathbf{j} \times \mathbf{v}$. Thus, it is verified that such an element has 4 instantaneous dof, 2 translations and 2 rotations.

The analysis of the chain 2 is identical, while chains 3 and 4 provide full mobility to the platform and therefore do not add any constraint.

So, once assembled the four kinematic chains with the end-effector, its motion capability is given by the intersection of twists generated by each kinematic chain. In other words, the end-effector resulting wrench is given by the union of all chains' wrenches. Like this, the wrench and the twist resulting in the moving platform are those defined by the chain 1 and determine the platform mobility.

2.1 Kinematic Analysis

The position problem is not solved in this paper due to lack of space, but since the motion is slow, it is possible to apply a numerical procedure in a practical way. The most representative properties of the mechanism motion are obtained from the velocity analysis. In order to do this, firstly equations due to constraints must be taken into account. From the approach presented in the previous section, it is established that in chains $i = 1, 2$ point P moves in the XZ plane, so its velocity does not have component along the Y axis and also moving platform angular velocity does not have component along the direction of its normal:

$$\dot{\mathbf{p}} \cdot \mathbf{j} = 0 \qquad \boldsymbol{\omega} \cdot [\mathbf{j} \times (\mathbf{a}_1 - \mathbf{p})] = 0 \tag{5}$$

When this is written in matrix format the constraint Jacobian is obtained:

$$\mathbf{J}_C \dot{\mathbf{x}} = \mathbf{0} \qquad \mathbf{J}_C = \begin{bmatrix} \mathbf{0} & \mathbf{j}^T \\ [\mathbf{j} \times (\mathbf{c}_1 - \mathbf{p})]^T & \mathbf{0} \end{bmatrix} \tag{6}$$

Alternatively, output kinematic screw terms $\dot{\mathbf{x}}$ can be reduced to the four velocity independent parameters $\dot{\mathbf{s}}$. Considering the existing constraints, the velocity of P can be expressed by means of just two components:

$$\dot{\mathbf{p}} = \dot{x}\mathbf{i} + \dot{z}\mathbf{k} \tag{7}$$

and also the angular velocity of the moving platform:

$$\boldsymbol{\omega} = \dot{\theta}\mathbf{j} + \dot{\psi}\frac{\mathbf{c}_1 - \mathbf{p}}{r} \tag{8}$$

where r is the length of vector $\mathbf{c}_1 - \mathbf{p}$

Therefore, $\dot{\mathbf{x}}$ can be rewritten as:

$$\dot{\mathbf{x}} = \begin{bmatrix} \boldsymbol{\omega} \\ \dot{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{c}_1 - \mathbf{p}}{r} & \mathbf{j} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{i} & \mathbf{k} \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{x} \\ \dot{z} \end{bmatrix} = \mathbf{J}_r \dot{\mathbf{s}} \tag{9}$$

Secondly, the loop equation of each kinematic chain has to be posed. Vector $(\mathbf{c}_i - \mathbf{b}_i)$, which has a constant modulus, can be isolated:

$$(\mathbf{c}_i - \mathbf{b}_i) = \mathbf{p} + \mathbf{R}(\mathbf{c}_i - \mathbf{p}) - \mathbf{a}_i - (\mathbf{b}_i - \mathbf{a}_i), \quad i = 1, \ldots, 4 \tag{10}$$

and the derivatives are:

$$\boldsymbol{\omega}_i \times (\mathbf{c}_i - \mathbf{b}_i) = \dot{\mathbf{p}} + \boldsymbol{\omega} \times (\mathbf{c}_i - \mathbf{p}) - \dot{\rho}_i \mathbf{k} \tag{11}$$

where $\boldsymbol{\omega}_i$ is the angular velocity of the rigid element B_iC_i.

In order to cancel those terms corresponding with passive velocities the dot product with $(\mathbf{c}_i - \mathbf{b}_i)$ is used, which yields:

$$\dot{\rho}_i \left[(\mathbf{c}_i - \mathbf{b}_i) \cdot \mathbf{k}\right] = (\mathbf{c}_i - \mathbf{b}_i) \cdot \dot{\mathbf{p}} + \boldsymbol{\omega} \cdot \left[(\mathbf{c}_i - \mathbf{p}) \times (\mathbf{c}_i - \mathbf{b}_i)\right] \tag{12}$$

Using $\mathbf{v}_i = (\mathbf{c}_1 - \mathbf{b}_1)/L_i$, Jacobian matrices are stated:

$$\mathbf{J}_x = \begin{bmatrix} \left[(\mathbf{c}_1 - \mathbf{p}) \times \mathbf{v}_1\right]^T & \mathbf{v}_1^T \\ \left[(\mathbf{c}_2 - \mathbf{p}) \times \mathbf{v}_2\right]^T & \mathbf{v}_2^T \\ \left[(\mathbf{c}_3 - \mathbf{p}) \times \mathbf{v}_3\right]^T & \mathbf{v}_3^T \\ \left[(\mathbf{c}_4 - \mathbf{p}) \times \mathbf{v}_4\right]^T & \mathbf{v}_4^T \end{bmatrix} \tag{13}$$

$$\mathbf{J}_q = \begin{bmatrix} [\mathbf{v}_1 \cdot \mathbf{k}] & 0 & 0 & 0 \\ 0 & [\mathbf{v}_2 \cdot \mathbf{k}] & 0 & 0 \\ 0 & 0 & [\mathbf{v}_3 \cdot \mathbf{k}] & 0 \\ 0 & 0 & 0 & [\mathbf{v}_4 \cdot \mathbf{k}] \end{bmatrix} \tag{14}$$

In this case, the Jacobian $\mathbf{J}_r$ (Eq. 9) can be used to obtain the velocity equation, reduced to independent terms $\dot{\mathbf{s}}$. So, the terms in $\dot{\mathbf{x}}$ can be substituted in Eq. 12 in order to obtain the velocity equation:

$$\mathbf{J}_x \mathbf{J}_r \dot{\mathbf{s}} = \mathbf{J}_q \dot{\mathbf{q}} \tag{15}$$

If the matrix premultiplying $\dot{\mathbf{s}}$ is inverted, and this result is premultiplied by $\mathbf{J}_q$, the Jacobian $\mathbf{J}^r$ is obtained. This can be now partitioned into the rotational and translational components of $\dot{\mathbf{s}}$ and their kinematic indicators of dexterity obtained.

$$\dot{\mathbf{s}} = \mathbf{J}^r \dot{\mathbf{q}} = \begin{bmatrix} \mathbf{J}_R^r \\ \mathbf{J}_T^r \end{bmatrix} \dot{\mathbf{q}} \tag{16}$$

These Jacobians have lower dimensions, and thus a lower computational cost. It is necessary to consider the homogenization of matrix $\mathbf{J}_x$. In order to achieve this, an isotropic posture can be used to define the characteristic length, or a temporary homogenization length can be used.

3 Final Design and operation

The solution proposed consists in raising a moving platform up to a certain height and let it fall by gravity effect in a controlled way. During the insolation hours, the platform has to be oriented to follow the sun. Once the target orientation is defined, the positioning of the moving platform must be obtained dropping some of the existing sliders, in order to move them

by the gravity effect. With this objective, an optimization of the geometric parameters of the mechanism has been carried out, in which a restriction was that actuators must move downwards in the solar tracking trajectory. With such a design, the main energy consumption of this system takes place when raising the platform, while the target descent motion is achieved acting on a set of mechanical brakes controlled electronically. The raising motion is done during the night, when the electrical energy is cheaper. The final design of the Solar Tracker is shown in Figure 3.

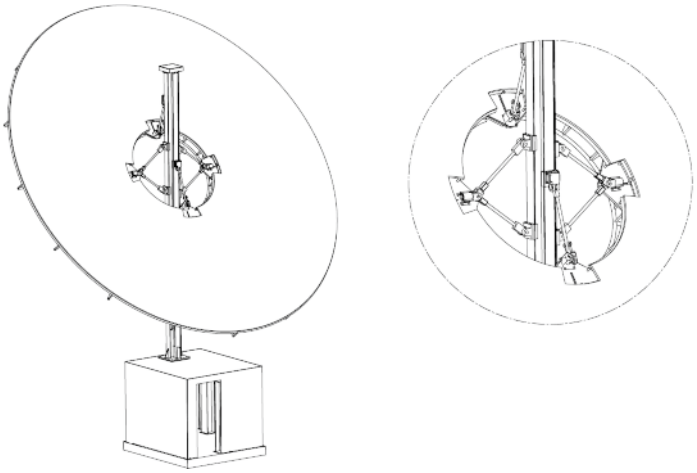


Figure 3. Final Design of the 2R2T Solar Tracker.

Then, the system is able to change its orientation continuously to follow the sun motion letting go the brakes of the sliders electronically. This operation involves minimum energy consumption and allows a high positioning accuracy. The final fall takes the mechanism to the flat orientation required to raise again the system during the night. Figure 4 shows the operating cycle of the 2R2T Solar Tracker.

4 Conclusion

This paper presents the design of a solar tracker based in parallel kinematics. The architecture of the mechanism allows the collector to maintain the orientation perpendicular to the direct solar radiation simply acting on a system of electronically controlled mechanical brakes. Thus, the main energy consumption takes place when raising the platform during the night and the total energy consumption is minimized.

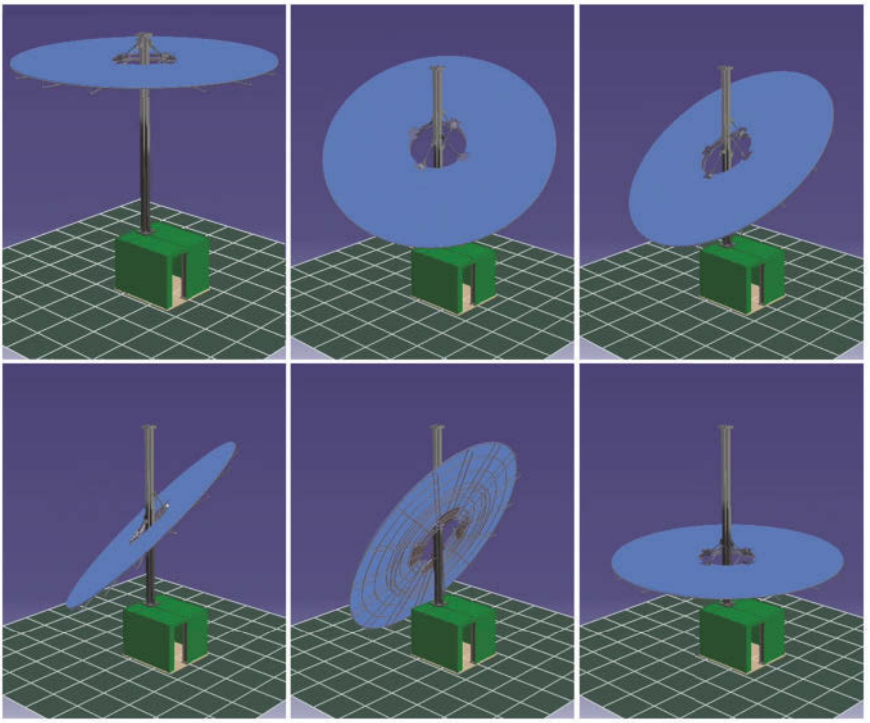


Figure 4. 2R2T Solar Tracker operating cycle.

Bibliography

C.S. Chin, A. Babu, and W. McBride. Design, modeling and testing of a standalone single axis active solar tracker using matlab/simulink. *Renewable Energy*, 36:3075–3090, 2011.

E. Lorenzo, M. Pérez, A. Ezpeleta, and J. Acedo. Design of tracking photovoltaic systems with a single vertical axis. *Progress in Photovoltaics: Research and Applications*, 10:533—-543, 2002.

J-P. Merlet. *Parallel Robots*. Springer, 2006.

Hossein Mousazadeh, Alireza Keyhani, Arzhang Javadi, Hossein Mobli, Karen Abrinia, and Ahmad Sharifi. A review of principle and sun-tracking methods for maximizing solar systems output. *Renewable and Sustainable Energy Reviews*, 13:1800—1818, 2009.

Calibration of a Fully-Constrained Parallel Cable-Driven Robot

Julien Alexandre dit Sandretto[*†], David Daney[†] and Marc Gouttefarde[§]
[*†]COPRIN INRIA, [§]LIRMM
INRIA, 2004 route des Lucioles, B.P. 93, 06902 Sophia Antipolis, France
{David.Daney, Julien.Alexandre_dit_Sandretto}@inria.fr

Abstract An identification of the model parameters for a parallel cable-driven robot (8 cables for 6 degrees of freedom) is performed by using both a calibration and a self-calibration approach. Additionally, advanced tools and algorithmic improvements are presented to perform the parameter identification. A complete experimentation validates the robot accuracy improvement.

1 Introduction

Because of tolerances in manufacturing or assembly, the geometry of the actual manipulator does not correspond to the desired design and its theoretical kinematic model. Consequently, the performances of the manipulator are reduced if not lost. Calibration consists in identifying model parameters through redundant information on the state of the robot generally provided by internal/external measurements. The more simple and common approach to calibrate a parallel robot is the implicit method as presented in Wampler et al. (1995). In the self-calibration case, the necessary data are provided by additional internal sensor(s). Many solutions have been proposed for parallel manipulators, and some of them Patel and Ehmann (2000); Takeda et al. (2004) may be easily adapted to the case dealt with in the present paper.

Cable-driven robots have several interesting properties like reduced mass of moving parts (for cables of negligible mass), ease of reconfiguration and, especially, a potentially very large workspace. They are notably used for a flying camera system sky (2007), and have been proposed for heavy loads transportation, for orienting heavy devices and for contour crafting Bosscher and al. (2007). In our case, the robot is actuated by eight cables for six DOF. Several studies have been run on cable-driven robot kinematics, but few concerning their calibration. However specific procedure are presented in Tadokoro et al. (1999) and in Varziri and Notash (2007) respectively.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_10,

The basic identification method is the non linear least squares approach, which computes the parameters so as to match model estimations with measures. Similar methods are orthogonal distance regression (ODR) Boggs et al. (1987) and χ^2 used in Patel and Ehmann (2000). Different approaches have been proposed like filtering adapted in Wampler et al. (1995), or an original interval approach proposed in Daney et al. (2006).

2 Cable-driven robot

This study is part of a project named CoGiRo (Control of Giant Robot) which notably aims at designing a parallel cable robot having $n = 6$ degrees of freedom and a very large workspace. It uses $m = 8$ cables controlling the 6 DOF motion of its mobile platform and its geometry has been chosen so that the platform is fully constrained by the cables. The moving platform or end-

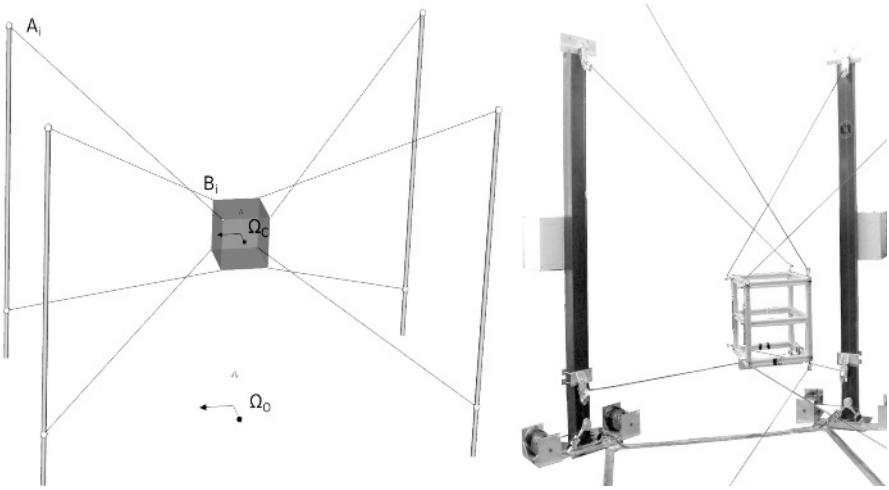


Figure 1. A cable-driven robot sketch and ReelAx8 picture

effector (mobile reference frame Ω_C) is connected to the base (fixed reference frame Ω_O). The i^{th} cable connects the point A_i of the base (coordinate a_i in Ω_O) to the point B_i on the mobile platform (coordinate b_i in Ω_C). The pose of the mobile (defined by the position P and the orientation R of Ω_C expressed in Ω_O) is directly controlled by the length and the tension of each cable. The implicit kinematic system of equations is given by

$$||P + Rb_i - a_i||^2 - L_i^2 = 0, \quad i = [1...m] \tag{1}$$

where L_i is the distance A_iB_i.

3 Calibration

The calibration goal is to enhance the robot performances by improvement of model knowledge.

We will see that calibration can be considered as a generic process Wampler et al. (1995). We will make a difference between the case where we have additional external measures on the state of the robot and the case where the proprioceptive sensor data of the robot are sufficient for calibration (also called self-calibration).

The robots studied, $m > n$, are redundant in terms of measurement if we make the hypothesis of non elastic and mass-less cables.

3.1 Generic view

Based on Wampler et al. (1995), for each of the N_C measure configuration the calibration equations links three types of variables:
The measurements M_k ($k = 1..N_C$), the parameters ξ we want to identify (geometrical parameters) and unknowns variables Υ required to model our equations. These variables $\Upsilon = [\hat{\Upsilon}, \check{\Upsilon}_{1..N_C}]$ should be

- Constant $\hat{\Upsilon}$: their values do not change during the calibration process;
- Variable as a function of the robot configurations $\check{\Upsilon}_{k=1..N_C}$.

We consider a system of equations linking a set of measures M and the unknowns $V = [\xi, \Upsilon]$ in the calibration equations:

$$f_k(M_k, V) \simeq 0, \quad k = [1...N_C] \tag{2}$$

A solution of the system (2) could be computed by different methods, most of them give a non linear least squares solution which minimizes the criteria $F^T.F$ with $F = [f_1, \ldots, f_{N_c}]^T$. This could be obtained with a classical Levenberg-Marquart algorithm but some improvement on criterion definition are possible.

Weighted Least Squares technique introduces the ability to prioritize a measure by considering the criterion $F^T \Sigma_F F$. The weight matrix Σ_F is built as a function of knowledge on some uncertainties linked with the measurement modeled through a covariance matrix Σ_M. A linear approximation of Σ_F is obtained as $\Sigma_F = J_M^T \Sigma_M J_M$ with $J_M = \frac{\partial F}{\partial M}$.

Orthogonal Distance Regression takes into account the possible errors in measurements Boggs et al. (1987) and considers the criteria $F^T \Sigma_F F + \tilde{M}^T \Sigma_M \tilde{M}$. We put $\tilde{M}$ as the difference between the current M and the initial M.

χ^2 permits a control of deviation in identification of the unknowns Patel and Ehmann (2000). The considered criteria is now $F^T \Sigma_F F + \tilde{M}^T \Sigma_M \tilde{M} + \tilde{V}^T \Sigma_V \tilde{V}$. We put $\tilde{V}$ as the difference between the current V and the initial V.

For the parallel cable robot calibration, the equations used are directly the kinematic relationships (1) provided that the hypothesis of negligible cable elasticity and mass is acceptable:

$$f_{k,i}(M_k, V) = ||P_k + R_k b_i - a_i||^2 - (\rho_{k,i} + \Delta l_i)^2 = 0$$

for $k = 1...N_C$ and $i = 1...m$.

Now let's discuss two different calibration approaches, with and without external measurements.

Calibration with external measures In addition to the articular coordinates given by the proprioceptive sensors, the measurement of the robot pose (position and orientation) provided by an external device like a camera or a laser tracker is assumed to be available. The calibration system to be solved is made of the functions $f_{k,i}(M_k, V)$ with the following data: $M_k = [\rho_{i,k}, P_k, R_k]$, $\xi = [a_i, b_i, \Delta l_i]$ and $V = [\Upsilon, \xi] = [\emptyset, \xi] = \xi$.

Self-calibration without external measures If we don't have any external measurement, we can calibrate the robot with the proprioceptive sensors only. The calibration system to be solved is still made of the functions $f_{k,i}(M_k, V)$ but with the following data: $M_k = [\rho_{i,k}]$, $\xi = [a_i, b_i, \Delta l_i]$ and $V = [\Upsilon, \xi] = [P_k, R_k, \xi]$.

With these data, the Jacobian $J_V = \frac{\partial f}{\partial V}$ is composed of the Jacobian of kinematics parameters (as in a calibration case) J_ξ and of the inverse kinematics Jacobian J_Υ.

One difficulty of calibration is to eliminate the $\breve{\Upsilon}_k = [P_k, R_k]$ variables Daney (2000) in the identification vector $V = [\breve{\Upsilon}_k, \xi]$. In Patel and Ehmann (2000), it is done indirectly with an iterative Forward Kinematics (FK) in order to determine Υ in each iteration of the identification algorithm.

We propose a complete identification which looks for Υ together with ξ. This allows us to avoid the problem of the FK convergence.

4 Experiments

ReelAx8, shown in Fig. 1, is a reconfigurable cable driven robot. Eight cables, wound round winches, are each attached to the eight corners of a cube shaped platform of about 40 centimeters, by means of spherical joints. The winches are fixed 2 by 2 on four posts up to three meters arranged at the four corners of a three by four meters rectangle.

4.1 Measurement

The measurements were made by means of a laser tracker system and a portable 3D measuring arm.

The acquisition of the measurements were made in 2 different steps. First, we found the estimation of geometrical parameters. We measured the eyelet positions on the frame with the laser tracker and the attachment points on the mobile platform with a portable 3D measuring arm. The pose measurement step could then start. We placed the mobile platform in 44 different poses and took measures of three types:

- Proprioceptive sensors gave cables lengths ;
- The robot force sensors gave us the tension in the cables (we don't use these data in this paper) ;
- Positions of three points measured with the laser gave us the position and orientation of the mobile platform.

A measure is useful if its precision is known. Having this in mind, we estimated the expected error for each device. This error was used in the verification step and in the computation of the weights of the identification process. In our case we considered a mix between the precision of measurement device and the way the acquisition was done(maximized on purpose): $\sigma_{poses} = 5mm$, $\sigma_{length} = 5mm$, $\sigma_A = 20mm$, $\sigma_B = 10mm$, $\sigma_{\Delta l} = 100mm$.

4.2 Experimental results

From the 44 measures done, we used 30 measures for the identification and 10 for validation (4 outliers were eliminated). For the self-calibration, 30 poses are sufficient and reduce the measurement noise effect. In our calibration study and with the goal of adapting it on a giant robot, we made an important work investment to obtain well-estimated kinematic parameters (by laser and CMM measurements) in order to check the robustness of the algorithms used.

Initially, we identified the 56 parameters $[a_i, b_i, \Delta l_i]_{i=1..8}$ but we found a strong dependence between the a_i and the b_i. This came from the small rotations allowed by the prototype. The parameters a_i and b_i are linked by the relation $R_k b_i - a_i$ in the identification equations with $R_k \simeq I_{3*3}, k = 1..N_C$. It's not an important problem, indeed, the parameters of the platform are well known, easy to measure and don't change unlike Δl_i and a_i, which changes at each new configuration, restarting of motors, etc. In the particular case of self-calibration, it's necessary to choose the reference frame of the robot as follows Besnard and Khalil (2001) : $a_{1x} = a_{1y} = a_{1z} = a_{2y} = a_{2z} = a_{3z} = 0$.

The results of the **calibration** which consists in the identification of the 32 parameters $[a_i, \Delta l_i]_{i=1..8}$ with external measures (expressed with the

	WLS		ODR	
Identification (30 configurations)				
Initial err	**mean(std)**		**mean(std)**	
	20mm (20mm)		20mm (20mm)	
Final err	3mm (6mm)		1mm (4mm)	
Validation (10 configurations)				
Err on calib eq	**initial** 40mm	**final** 6mm	**initial** 40mm	**final** 9mm
Err positioning	37%	9%	37%	10%
Error orienting	40.2%	25%	40.2%	21%

Table 1. Results for the calibration

	WLS		χ^2	
Identification (30 configurations)				
Initial err	**mean(std)**		**mean(std)**	
	20mm (30mm)		20mm (30mm)	
Final err	0.5mm (0.5mm)		2mm (4mm)	
Validation (10 configurations)				
Error on calib eq	**initial** 40mm	**final** 1mm	**initial** 40mm	**final** 0.7mm
Err positioning	37%	1.2%	37%	0.3%
Error orienting	40.2%	12%	40.2%	1.9%

Table 2. Results for the self-calibration

residual error on the cable lengths between measures and inverse model), are collected in the first part of table 1.

The results of the **self-calibration** which consists in the identification of the 26 parameters $[a_{2x}, a_{3x}, a_{3y}, a_i]_{i=4..8}$ and $[\Delta l_i]_{i=1..8}$ simultaneously to the N_C pose estimation, without external measures (residual error on the cable lengths), are collected in the first part of table 2.

For the both method, after 4 iterations we reach a correct minimum (with a medium error at 0.5mm for WLS and 2mm for χ^2) and after about 10 iterations the solver stops at the expected precision ($\Delta F < 10^{-8}$).

First, a simple validation is done by the checking of model improvement on the 10 validation measures. The results (residual error on the cable lengths) are then presented in the second part of table 1 for calibration and table 2 for self-calibration.

Second, we validate the manipulator accuracy by computing the improvement of general positioning. We measure a moving (difference) between two validation configurations, on position ΔP_{meas}, and on orientation

ΔR_{meas} (in Euler angles). We compute the theoretical moving ΔP_λ and the rotation ΔR_λ with a FK process, where λ means the kind of kinematic parameters used in the FK (i.e initial, post WLS calibration, post ODR, post WLS self-calibration, or post χ^2 identification). The results are given as relative error in percentages and defined by $100 * (\frac{\Delta P_\lambda - \Delta P_{meas}}{\Delta P_{meas}})$.

A graphical result for one displacement is shown in detail for the positioning and for the orienting in Fig. 2 .

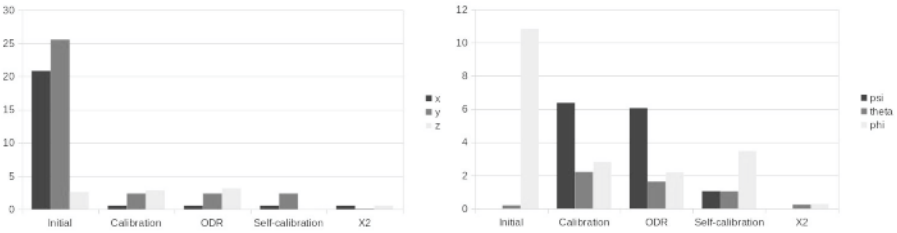


Figure 2. Error on positioning and orienting % (for initial parameters and after 4 different identification methods)

The complete results are expressed with a maximal error on P and R given in percentages. They are shown in the lower part of table 1 for calibration and table 2 for self-calibration.

We see that the orienting is not perfectly corrected (except for the χ^2 method), this is due to the fact that we could not obtain measurements in a wide workspace at different orientations larger than 5 degrees.

5 Conclusions and future works

In this paper, we have verified experimentally the hypothesis of self-calibration capacity for a particular parallel cable-driven robot. Now those allow us to test a simple new approach for the elimination of pose variables in the self-calibration process. We try three different methods derived from the least squares approach for the parameter identification, and make some proposals on their use. To conclude, our robot can be either calibrated if we don't have accurate kinematics parameter estimation, or self-calibrate, both with robust algorithms.

The robot under construction for the CoGiRo project will have a different geometry and different cables to handle heavy loads; for that, a new model is in progress with mass and elasticity consideration. Future works will include certification of the identification results. In the project, we plan to use camera for 3D pose sensing, and we are developing a calibration method

based on the vision result.

Acknowledgments

This work was supported for one part by the French National Research Agency (ANR) under grant 2009-SEGI-018 (CoGiRo ANR project) and for the other part by the Région Languedoc-Roussillon under grant 115217. Thanks to Gilles Trombettoni for fruitful discussions.

Bibliography

Skycam, 2007. URL `www.skycam.tv`.

Sebastian Besnard and Wisama Khalil. Identifiable parameters for parallel robots kinematic calibration. In *International Conference on Robotics and Automation*, volume 3, pages 2859–4729, Seoul, Korea, may 2001.

Paul T. Boggs, Richard H. Byrd, and Robert B. Schnabel. A stable and efficient algorithm for nonlinear orthogonal distance regression. *SIAM Journal on Scientific and Statistical Computing*, 8(6):1052–1078, 1987.

Paul Bosscher and al. Cable-suspended robotic contour crafting system. *Automation in Construction*, 17(1):45 – 55, 2007.

David Daney. *Etalonnage geometrique des robots paralleles*. PhD thesis, Nice - Sophia Antipolis, 2000.

David Daney, Nicolas Andreff, Gilles Chabert, and Yves Papegay. Interval method for calibration of parallel robots : Vision based experiments. *Mechanism and Machine Theory*, 41:929–944, 2006.

Amit J. Patel and Kornel F. Ehmann. Calibration of a hexapod machine tool using a redundant leg. *International Journal of Machine Tools and Manufacture*, 40(4):489–512, mar 2000.

S. Tadokoro, R. Verhoeven, M. Hiller, and T. Takamori. A portable parallel manipulator for search and rescue at large-scale urban earthquakes and an identification algorithm for the installation in unstructured environments. In *Intelligent Robots and Systems, IROS '99.*, volume 2, pages 1222 –1227 vol.2, 1999.

Yukio Takeda, Gang Shen, and Hiroaki Funabashi. A DBB-Based kinematic calibration method for In-Parallel actuated mechanisms using a fourier series. *Journal of Mechanical Design*, 126(5):856–865, 2004. doi: 10.1115/1.1767822.

M. Saeed Varziri and Leila Notash. Kinematic calibration of a wire-actuated parallel robot. *Mechanism and Machine Theory*, 42(8):960 – 976, 2007.

Charles W. Wampler, John M. Hollerbach, and Tatsuo Arai. An implicit loop method for kinematic calibration and its application to Closed-Chain mechanisms. *IEEE Transactions on Robotics and Automation*, 11 (5):710–724, oct 1995.

Global Identification of Drive Gains and Dynamic Parameters of Parallel Robots - Part 1: Theory

Sébastien Briot[‡] and Maxime Gautier [†‡*]

[†] LUNAM, Université de Nantes, Nantes, FRANCE
[‡] Institut de Recherche en Communications et Cybernétique de Nantes
IRCCyN, UMR CNRS 6592, Nantes, FRANCE

Abstract Most of the papers dealing with the dynamic parameters identification of parallel robots are based on simple models, which take only the dynamics of the moving platform into account. Moreover the actuator drive gains are not calibrated which leads to identification errors. In this paper a systematic way to derive the full dynamic identification model of parallel robots is proposed in combination with a method that allows the identification of both robot inertial parameters and drive gains.

1 Introduction

Parallel robots are increasingly being used since a few decades. This is due to their main advantages compared to their serial counterparts that are: (i) a higher intrinsic rigidity, (ii) a larger payload-to-weight ratio and (iii) higher velocity and acceleration capacities (Merlet, 2006). In order to obtain these interesting properties, a good controller should be implemented. Several approaches could be envisaged (Amiral et al., 1996; Vivas and Poignet, 2005), but it appears that, for high-speed robots or when varying loads have to be compensated (e.g. in pick-and-place operations or machining), computed torque control is generally used (Khalil and Dombre, 2002). This approach needs a correct identification of the dynamic model of the robot with the load (Khalil et al., 2007), which can be obtained provided two main conditions are satisfied: (i) a well-tuned derivative band-pass filtering of actuated joints position is used to calculate the actuated joints velocities and accelerations, and (ii) the accurate values of actuator drive gains g_τ are accurately known to calculate the actuator force/torque as the product of

[*]This work has been partially funded by the French ANR project ARROW (ANR 2011 BS3 006 01).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_11,

the known control signal calculated by the numerical controller of the robot (the current references) by the drive gains. However, this is rarely the case and they need to be calibrated (Restrepo and Gautier, 1995; Corke, 1996).

Among all papers dealing with dynamic parameters identificiation for parallel robots, only few of them propose a systematic computation of the full Inverse Dynamic Identification Model (IDIM). In (Guegan et al., 2003), the authors propose an attempt to create a systematic IDIM based on a Newton-Euler approach. The closed loops are first virtually opened to compute the dynamic model of the tree structure and then, the closure constraints are imposed. However, the way to open the loop is not straightforward. In (Grotjahn et al., 2004; Diaz-Rodriguez et al., 2010), the authors propose methods for computing the IDIM based on the Jourdain's principle or Lagrange multipliers. But the way to identify the drive gains is not treated. Moreover, some jacobian matrices, whose computation is not straightforward, are not clearly derived.

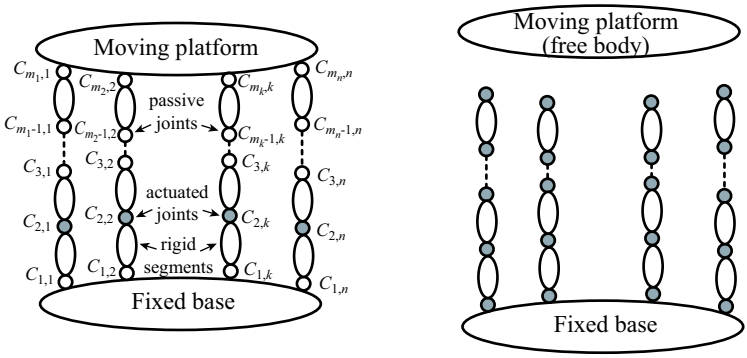
In this paper it is proposed a global approach for both the identification of parallel robots dynamic parameters and drive gains. This paper is the first part of our work on the identification of the parallel robots dynamic parameters and it presents the theoretical approach. A case study will be treated in (Briot and Gautier, 2012).

2 A Systematic Procedure for the IDIM Computation

A parallel robot is a complex multi-body system having several closed loops (Fig. 1a). It is composed of a moving platform connected to a fixed base by n legs and m elements. In this paper, a method similar to (Ibrahim and Khalil, 2010) is applied for the computation of IDIM of *non-redundant* parallel robots. The proposed method is decomposed into two steps: (i) all closed loops are virtually opened to make the platform virtually disassembled from the rest of the structure (Fig. 1b); each leg joint is virtually considered actuated (even for unactuated actual joints) so that the robot becomes a tree structure with a free body: the platform; the dynamic model of the tree structure and of the free platform is then computed using a systematic procedure based on the Newton-Euler principle and (ii) the loops are then closed using the loop-closure equations and the Lagrange multipliers.

2.1 IDIM of Tree Open Loop Robots

It is known that the complete rigid dynamic model of any open-loop tree structure can be linearly written in term of a $(n_t \times 1)$ vector with respect



(a) Kinematic chain (b) Virtual tree structure

Figure 1: A general parallel robot.

to the standard parameters χ_{st_t} (Khalil and Dombre, 2002),

$$\tau_{idm_t}\left(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t\right) = \phi_{st_t}\left(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t\right)\chi_{st_t} \tag{1}$$

where τ_{idm_t} is the $(n_t \times 1)$ vector of the virtual input efforts of the tree structure, ϕ_{st_t} is the $(n_t \times n_{st_t})$ jacobian matrix of τ_{idm_t}, with respect to the $(n_{st_t} \times 1)$ vector χ_{st_t} of the standard parameters given by $\chi_{st_t} = [\chi_{st}^{1T}, \chi_{st}^{2T}, \ldots \chi_{st}^{nT}]$ that are described in (Khalil and Dombre, 2002) and $\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t$ are the vectors of the joint positions, velocities and accelerations, respectively.

Several methods can be used to systematically derive these equations. Here, an algorithm based on the use of the modified Denavit-Hartenberg robot geometric description and the Newton-Euler principle is applied. This modelling is known to give the dynamic model equations in the most compact form (Khalil and Dombre, 2002).

2.2 IDIM of Parallel Robots

The previous dynamic model does not take into account the closed loop characteristics of parallel robots: among all joint coordinates $\mathbf{q}_t$ and platform coordinates $\mathbf{x}$, only a subset denoted as $\mathbf{q}$ is independent (the actual actuated joints positions). As a result, vectors $\mathbf{q}_t$ and $\mathbf{x}$ can be computed as functions of $\mathbf{q}$ using the loop-closure equations (Merlet, 2006),

$$\mathbf{f}_t(\mathbf{q}, \mathbf{q}_t) = \mathbf{0}, \mathbf{f}_p(\mathbf{q}, \mathbf{x}) = \mathbf{0} \tag{2}$$

Using these equations, all joint and platform velocities and accelerations

can be computed:

$$\dot{\mathbf{q}}_t = -\mathbf{A}_t^{-1}\mathbf{B}_t\dot{\mathbf{q}} = \mathbf{J}_t\dot{\mathbf{q}}, \mathbf{t} = -\mathbf{A}_p^{-1}\mathbf{B}_p\dot{\mathbf{q}} = \mathbf{J}_p\dot{\mathbf{q}}, \tag{3}$$

$$\ddot{\mathbf{q}}_t = -\mathbf{A}_t^{-1}(\dot{\mathbf{A}}_t\dot{\mathbf{q}}_t + \mathbf{B}_t\ddot{\mathbf{q}} + \dot{\mathbf{B}}_t\dot{\mathbf{q}}), \gamma = -\mathbf{A}_p^{-1}(\dot{\mathbf{A}}_p\mathbf{t} + \mathbf{B}_p\ddot{\mathbf{q}} + \dot{\mathbf{B}}_p\dot{\mathbf{q}}) \tag{4}$$

where $\mathbf{t}$ is the platform twist, γ the platform acceleration screw and matrices $\mathbf{A}_t$, $\mathbf{A}_p$ ($\mathbf{B}_t$, $\mathbf{B}_p$, resp.) can be obtained through the differentation of the loop-closure equations (2) with respect to all joint coordinates $\mathbf{q}_t$ and the platform coordinates (actuated joints positions, resp.). It should be noticed that the computation of matrices $\mathbf{A}_t$ and $\mathbf{B}_t$ is generally not straightforward. Therefore, it is preferable to:

1. express the kinematic relation between the platform twist $\mathbf{t}$ and the velocities $\mathbf{t}_{tk}$ of all leg extremities $C_{m_k,k}$ (Fig. 1a), $\mathbf{t}_{tk} = \mathbf{J}_{tk}\mathbf{t}$,
2. express the kinematic relation between the velocities $\mathbf{t}_{tk}$ of all leg extremities $C_{m_k,k}$ and the velocities of all joints $\dot{\mathbf{q}}_p$, $\mathbf{t}_{tk} = \mathbf{J}_k\dot{\mathbf{q}}_p$,
3. combine these two relations with (3) in order to obtain $\dot{\mathbf{q}}_p = \mathbf{J}_t\dot{\mathbf{q}}$, with $\mathbf{J}_t = \mathbf{J}_k^{-1}\mathbf{J}_{tk}\mathbf{J}_p$. All the previous expressions are valuable as long as the robot does not meet any singularity.

To take into account the closure-loop constraints into the dynamic model of the parallel robot, the Lagrange multipliers λ can be used (Khalil and Dombre, 2002) to compute the $(n \times 1)$ vector of the actuated joint force/ torque τ_{idm} of the closed-loop structure:

$$\tau_{idm} = -[\mathbf{B}_t^T, \mathbf{B}_p^T]\lambda, \text{ where } \begin{bmatrix} \mathbf{A}_t^T & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_p^T \end{bmatrix} \lambda = \begin{bmatrix} \tau_{idm_t} \\ \mathbf{f}_p \end{bmatrix} \tag{5}$$

$\mathbf{f}_p$ being the (6×1) vector of inertia forces of the platform plus the external loading (Ibrahim and Khalil, 2010). (5) can be rewritten as:

$$\begin{aligned} \tau_{idm} &= \mathbf{J}_t^T\tau_{idm_t} + \mathbf{J}_p^T\mathbf{f}_p = \mathbf{J}_t^T\phi_{st_t}(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t)\chi_{st_t} + \mathbf{J}_p^T\phi_p(\mathbf{x}, \mathbf{v}, \gamma)\chi_p \\ &= \begin{bmatrix} \mathbf{J}_t^T\phi_{st_t} & \mathbf{J}_p^T\phi_p \end{bmatrix} \begin{bmatrix} \chi_{st_t}^T & \chi_p^T \end{bmatrix}^T = \phi_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\chi_{st} \end{aligned} \tag{6}$$

where χ_p is the (10×1) vector of platform parameters, ϕ_p is the (6×10) jacobian matrix between $\mathbf{f}_p$ and χ_p, χ_{st} is the $(n_{st} \times 1)$ vector of the global standard parameters of the parallel robot and ϕ_{st} the $(n \times n_{st})$ jacobian matrix between τ_{idm} and χ_{st}.

The identifiable parameters are the base parameters which are the minimum number of dynamic parameters from which the dynamic model can be calculated (Khalil and Dombre, 2002). The minimal dynamic model can be written using the n_b base dynamic parameters χ as follows:

$$\tau_{idm} = \phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\chi \tag{7}$$

where ϕ is a subset of independant columns in ϕ_{st} which defines the identifiable parameters.

Because of perturbations due to noise measurement and modelling errors, the actual force/torque τ differs from τ_{idm} by an error, $\mathbf{e}$, such that:

$$\tau = \mathbf{v}_\tau \mathbf{g}_\tau = diag(v_\tau^j) \begin{bmatrix} g_\tau^1 & \dots & g_\tau^n \end{bmatrix}^T = \tau_{idm} + \mathbf{e} = \phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\chi + \mathbf{e} \quad (8)$$

$\mathbf{v}_\tau$ is the $(n \times n)$ diagonal matrix of the actual current references of the current amplifiers (v_τ^j corresponds to actuated joints j) and $\mathbf{g}_\tau$ is the $(n \times 1)$ vector of the drive gains (g_τ^j corresponds to actuator j). Equation (8) represents the IDIM.

3 Identification Procedure

3.1 Recalls on Least Squares Identification of the Dynamic Parameters (IDIM-LS)

The off-line identification of the base dynamic parameters χ can be achieved given measured or estimated off-line data for τ and $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, collected while the robot is tracking some planned trajectories. The model (8) is sampled, low pass filtered and decimated in order to get an overdetermined linear system of $(n \times r)$ equations and n_b unknowns:

$$\mathbf{Y}(\tau) = \mathbf{W}(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}, \hat{\ddot{\mathbf{q}}})\chi + \rho \quad (9)$$

where $(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}, \hat{\ddot{\mathbf{q}}})$ are an estimation of $(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, obtained by band-pass filtering and sampling the measure of $\mathbf{q}$ (Gautier, 1997), ρ is the $(r \times 1)$ vector of errors and $\mathbf{W}(\hat{\mathbf{q}}, \hat{\dot{\mathbf{q}}}, \hat{\ddot{\mathbf{q}}})$ is the $((n \times r) \times n_b)$ observation matrix.

Using the base parameters and tracking some 'exciting' reference trajectories (Gautier and Khalil, 1992), a well conditionned matrix $\mathbf{W}$ is obtained. The LS solution $\hat{\chi}$ of (9) is given by:

$$\hat{\chi} = \mathbf{W}^+\mathbf{Y}, \text{ where } \mathbf{W}^+ = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T \quad (10)$$

Standard deviations $\sigma_{\hat{\chi}_i}$ can be estimated assuming that ρ is a zero mean independant noise (Gautier, 1997). The ordinary LS can be improved by taking into account different standard deviations on actuated joint j equations errors (Gautier, 1997). This weighting operation normalises the errors in (9) and gives the weighted LS estimation of the parameters (IDIM-WLS).

3.2 Total Least Squares Identification

In the classical IDIM-LS, to compute vector $\mathbf{Y}$, the drive gains are supposed known. But usually the manufacturers give drive gain parameters

with an uncertainty of about 10%, thus leading to identification errors. Therefore, it is preferable to introduce the drive gains into the base parameters and to use the Total Least Squares Identification (IDIM-TLS).

Details on the Total LS (TLS) identification method can be found in (Huffel and Vandewalle, 1991) and many papers of the same authors. This method has been applied in (Gautier et al., 1994) for the identification of the drive gains and the dynamic parameters on a two degrees of freedom (dof) serial robot but gives arguable results due to the lack of an accurate scale factor. In this paper a major improvement is proposed: the accurate scaling of parameters using the precise weighed value of an additional payload mass. However, by the use of the model (9) without any modification, the payload parameters are regrouped with the end-effector parameters and cannot be independently identified. In order to apply the proposed approach, the model (9) must be modified. This procedure is detailed below.

IDIM Including a Payload and Drive Gains The inertial parameters of the payload are easily added to the IDIM by considering the payload as a link $m+1$ fixed to the robot platform. The model (8) becomes:

$$\tau = \mathbf{v}_\tau \mathbf{g}_\tau = \begin{bmatrix} \phi & \phi_{uL} & \phi_{kL} \end{bmatrix} \begin{bmatrix} \chi^T & \chi_{uL}^T & \chi_{kL}^T \end{bmatrix}^T + \mathbf{e} \tag{11}$$

where χ_{kL} is the $(n_{kL} \times 1)$ vector of the known inertial parameters of the payload (calculated with CAD or accurately measured), χ_{uL} is the $((10 - n_{kL}) \times 1)$ vector of the unknown inertial parameters of the payload, ϕ_{kL} is the $(n \times n_{kL})$ jacobian matrix of τ_{idm}, with respect to the vector χ_{kL} and ϕ_{uL} is the $(n \times (10 - n_{kL}))$ jacobian matrix of τ_{idm}, with respect to the vector χ_{uL}.

Solution of the IDIM-TLS The identification of the dynamic parameters of the robot and the payload requires the achievement of two types of trajectories: (a) trajectories without payload and (b) trajectories with the payload fixed to the end-effector (Khalil et al., 2007). The sampling and filtering of the model IDIM (11) can be then written as:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{V}_{\tau a} \\ \mathbf{V}_{\tau b} \end{bmatrix} \mathbf{g}_\tau = \begin{bmatrix} \mathbf{W}_a & \mathbf{0} & \mathbf{0} \\ \mathbf{W}_b & \mathbf{W}_{uL} & \mathbf{W}_{kL} \end{bmatrix} \begin{bmatrix} \chi^T & \chi_{uL}^T & \chi_{kL}^T \end{bmatrix}^T + \rho \tag{12}$$

where $\mathbf{W}_a$ is the observation matrix of the robot in the unloaded case, $\mathbf{W}_b$ is the observation matrix of the robot in the loaded case, $\mathbf{W}_{uL}$ is the observation matrix corresponding to the unknown payload inertial parameters, $\mathbf{W}_{kL}$ is the observation matrix corresponding to the known payload inertial parameters, $\mathbf{V}_{\tau a}$ is the matrix of $\mathbf{v}_\tau$ samples in the unloaded case, $\mathbf{V}_{\tau b}$ is the matrix of $\mathbf{v}_\tau$ samples in the loaded case.

Eq. (12) becomes:

$$\mathbf{W}_{tot}\chi_{tot} = \begin{bmatrix} -\mathbf{W}_a & \mathbf{V}_{\tau a} & \mathbf{0} & \mathbf{0} \\ -\mathbf{W}_b & \mathbf{V}_{\tau b} & -\mathbf{W}_{uL} & -\mathbf{W}_{kL}\chi_{kL} \end{bmatrix} \left[\chi^T, \mathbf{g}_\tau^T, \chi_{uL}^T, \delta\right]^T = \rho \quad (13)$$

where $\mathbf{W}_{tot}$ is a $(r \times (n_b + n + 11 - n_{kL}))$ matrix, χ_{tot} is a $(n_b+n+11-n_{kL})$ vector and δ is a scalar which should be equal to 1.

Without perturbation, $\rho = 0$ and $\mathbf{W}_{tot}$ should be rank deficient to get the solutions $\lambda\chi_{tot} \neq \mathbf{0}$ depending on a scale coefficient λ. However because of the measurement perturbations, $\mathbf{W}_{tot}$ is a full rank matrix. Therefore, the system (13) is replaced by the compatible system closest to (13) with respect to the Frobenius norm: $\hat{\mathbf{W}}_{tot}\hat{\chi}_{tot} = \mathbf{0}$, where $\hat{\mathbf{W}}_{tot}$ is the rank deficient matrix, with the same dimension as $\mathbf{W}_{tot}$, which minimizes the Frobenius norm $\|\mathbf{W}_{tot} - \hat{\mathbf{W}}_{tot}\|$ (Gautier et al., 1994) and $\hat{\chi}_{tot} = \left[\hat{\chi}^T \quad \hat{\mathbf{g}}_\tau^T \quad \hat{\chi}_{uL}^T \quad \hat{\delta}\right]^T$ is the solution of the compatible system closest to (13).

There are infinity of vectors $\hat{\chi}_{tot} = \lambda\hat{\chi}_{tot}^n$ that can be obtained by a scale factor λ. A unique solution $\hat{\chi}_{tot}^* = \hat{\lambda}\hat{\chi}_{tot}^n$ can be found by taking into account that the last value of $\hat{\chi}_{tot}^*$ should be equal to 1, i.e. $\hat{\lambda} = 1/\hat{\delta}$. More information on IDIM-TLS can be found in (Gautier and Briot, 2012).

This section ends the theoretical part of this work. A case study is developped in (Briot and Gautier, 2012) and shows the effectivess of the method.

4 Conclusions

This paper has presented a global approach for both the identification of parallel robots dynamic parameters and drive gains. It is based on a IDIM-TLS technique using current reference and position sampled data while the robot is tracking one reference trajectory without load fixed on the robot and one trajectory with a known payload fixed on the robot, whose inertial parameters are measured. This paper has presented only theoretical derivations and a case study will be presented in (Briot and Gautier, 2012).

Bibliography

Y. Amiral, G.F. Francois, J. Pontnauand, and M. Dafaoui. Design and control of a new six-dof parallel robot: application to equestrian gait simulation. *Mechatronics*, 6:227239, 1996.

S. Briot and M. Gautier. Global identification of drive gains and dynamic parameters of parallel robots - part 2: Case study. In *Proceedings of*

the 19th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control (RoManSy), 2012.

P. Corke. In situ measurement of robot motor electrical constants. *Robotica*, 23(14):433436, 1996.

M. Diaz-Rodriguez, V. Mata, A. Valera, and A. Page. A methodology for dynamic parameters identification of 3-dof parallel robots in terms of relevant parameters. *Mechanism and Machine Theory*, 45:13371356, 2010.

M. Gautier. Dynamic identification of robots with power model. In *Proceedings IEEE ICRA*, pages 1922–1927, Albuquerque, USA, April 1997.

M. Gautier and S. Briot. Global identification of drive gains parameters of robots using a known payload. In *Proceedings of the 2012 International Conference on Robotics and Automation (ICRA 2012)*, 2012.

M. Gautier and W. Khalil. Exciting trajectories for the identification of the inertial parameters of robots. *International Journal of Robotics Research*, 11(4):362–375, 1992.

M. Gautier, P. Vandanjon, and C. Presse. Identification of inertial and drive gain parameters of robots. In *Proceedings IEEE CDC*, pages 3764–3769, Lake Buena Vista, FL, USA, 1994.

M. Grotjahn, B. Heiman, and H. Abdellatif. Identification of friction and rigid-body dynamics of parallel kinematic structures for model-based control. *Multibody System Dynamics*, 11:273294, 2004.

S. Guegan, W. Khalil, and Ph. Lemoine. Identification of the dynamic parameters of the orthoglide. In *Proceedings IEEE ICRA*, pages 3272–3277, Taipei, Taiwan, September 2003.

S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Frontiers in Applied Mathematics series 9. Philadelphia, Pennsylvania: SIAM, 1991.

O. Ibrahim and W. Khalil. Inverse and direct dynamic models of hybride robots. *Mechanism and Machine Theory*, 45:627–640, 2010.

W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots.* Hermes Penton London, 2002.

W. Khalil, M. Gautier, and P. Lemoine. Identification of the payload inertial parameters of industrial manipulators. In *Proceedings IEEE ICRA*, pages 4943–4948, Roma, Italy, April 2007.

J.P. Merlet. *Parallel Robots*. Springer, 2nd edition, 2006.

P.P. Restrepo and M. Gautier. Calibration of drive chain of robot joints. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 526–531, 1995.

A. Vivas and P. Poignet. Predictive functional control of a parallel robot. *Control Engineering Practice*, 13:863874, 2005.

Global Identification of Drive Gains and Dynamic Parameters of Parallel Robots - Part 2: Case Study

Sébastien Briot[‡] and Maxime Gautier [†‡*]

[†] LUNAM, Université de Nantes, Nantes, FRANCE
[‡] Institut de Recherche en Communications et Cybernétique de Nantes
IRCCyN, UMR CNRS 6592, Nantes, FRANCE

Abstract Usually, identification models of parallel robots are simplified and take only the dynamics of the moving platform into account. Moreover the input efforts are estimated through the use of the manfucaturer's actuator drive gains that are not calibrated thus leading to identification errors. In this paper a systematic way to derive the full dynamic identification model of the Orthoglide parallel robot in combination with a method that allows the identification of both robot inertial parameters and drive gains.

1 Introduction

Many modern robotic applications require high-performances (in terms of speed, accuracy, payload capability, etc.) that cannot be obtained without a good controller (Amiral et al., 1996; Vivas and Poignet, 2005). It appears that, for high-speed robots or when varying loads have to be compensated (e.g. in pick-and-place operations or machining), computed torque control is generally used (Khalil and Dombre, 2002). This approach needs a correct identification of the dynamic model of the robot with the load (Khalil et al., 2007).

Several papers deal with the identification of parallel robots (Honegger et al., 2000, 1997). These publications are restricted to very simple models, which take only the dynamics of the moving platform into account. Only few of them deal with the systematic computation of the full Inverse Dynamic Identification Model (IDIM). In (Guegan et al., 2003), the authors propose an attempt to create a systematic IDIM based on a Newton-Euler approach. The closed loops are first virtually opened to compute the dynamic model of

*This work has been partially funded by the French ANR project ARROW (ANR 2011 BS3 006 01).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_12,

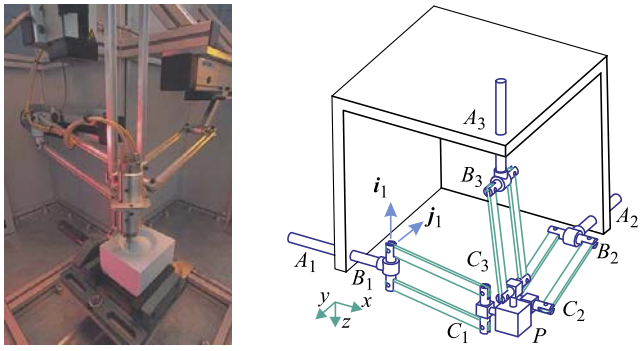
the tree structure and then, the closure constraints are imposed. However, the way to open the loop is not straightforward. In (Grotjahn et al., 2004; Diaz-Rodriguez et al., 2010), the authors propose methods for computing the IDIM based on the Jourdain's principle or Lagrange multipliers. But the way to identify the drive gains is not treated. Moreover, some jacobian matrices, whose computation is not straightforward, are not clearly derived.

This paper presents the application of a Total Least Square procedure for the identification of the inertial parameters (IDIM-TLS) of the Orthoglide (Chablat et al., 2004). It is the second part of our work on the identification of the parallel robots dynamic parameters and it uses the theoretical approach developped in (Briot and Gautier, 2012).

2 Modeling of the Orthoglide

2.1 Description of the architecture

The Orthoglide is a parallel robot with three translational degrees of freedom (dof) composed of three identical legs (Fig. 1). Each leg is achieved via one linearly actuated foot linked at its extremity to a spatial parallelogramm (Fig. 2a). The parallelogram is also attached to the mobile platform.

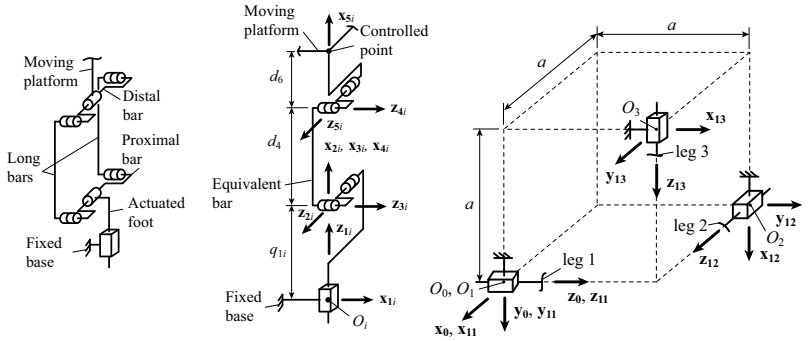


(a) Prototype (b) Kinematic chain

Figure 1: The Orthoglide robot.

The direction of the three linear actuators of the Orthoglide are orthogonal (Fig. 1b). This aims at creating a mechanism with a workspace shape close from a cube and whose behavior is close from the isotropy wherever it is located in its workspace.

For the remainder of the paper, it should be mentioned that the gravity field g is directed along the vector $\mathbf{y}_0$ of Fig. 2c.



(a) Kinematics of one leg (b) Equivalent kinematics of one leg (c) Fixed base

Figure 2: Kinematic chain description.

2.2 Dynamic modeling

For the dynamic modeling of the Orthoglide, as the bars of the parallelograms are parallel and as the prototype has been design so that they are identical, it is possible to replace the long bars by only one equivalent bar linked at each extremity by two orthogonal pivots (Fig. 2b) without loss of generality. Let us recall that the computation of the dynamic model is decomposed into two steps (Briot and Gautier, 2012):

1. All closed loops are virtually opened and the robot platform is virtually disassembled from the rest of the structure so that an open-loop tree structure is created;
2. The loops are then closed using the loop-closure equations and the Lagrange multipliers.

Dynamic modeling of the virtual tree structure Using the previous assemption, the Orthoglide kinematics necessary for computing the dynamic model are defined using the modified Denavit and Hartenberg notation (MDH) (Khalil and Dombre, 2002). The geometric parameters of the virtual open-loop tree structure are given in Tables 1 and 2. The MDH notation is well known, therefore the parameters of Tables 1 and 2 will not be defined here. For more information concerning the MDH parameters, the reader should refer to (Khalil and Dombre, 2002).

Using these MDH parameters, the dynamic model of the virtual tree structure that expresses the virtual input efforts τ_{idm_t} of all joints as a linear function of the inertial parameters χ_{st_t} can be automatically computed with

an algorithm based on the Newton-Euler approach (Khalil and Dombre, 2002; Khalil and Creusot, 1997):

$$\tau_{idm_t}(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t) = \phi_{st_t}(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t)\chi_{st_t} \tag{1}$$

where $\mathbf{q}_t = [q_{11}, q_{21}, ..., q_{53}]^T$ is the (15×1) vector regrouping all joint variables.

Table 1: MDH parameters for the frames corresponding to the first body of each leg.

j_i	$a(j_i)$	μ_{ji}	σ_{ji}	γ_{ji}	b_{ji}	α_{ji}	d_{ji}	θ_{ji}	r_{ji}
1_1	0	1	1	0	0	0	0	0	q_{11}
1_2	0	1	1	$\pi/2$	a	$\pi/2$	0	0	$q_{12} - a$
1_3	0	1	1	0	a	$-\pi/2$	0	$-\pi/2$	$q_{13} + a$

Table 2: MDH parameters for the frames corresponding to the leg i.

j_i	$a(j_i)$	μ_{ji}	σ_{ji}	γ_{ji}	b_{ji}	α_{ji}	d_{ji}	θ_{ji}	r_{ji}
2_i	1_i	1	0	0	0	$-\pi/2$	0	q_{2i}	0
3_i	2_i	1	0	0	0	$-\pi/2$	0	q_{3i}	0
4_i	3_i	1	0	0	0	0	d_4	q_{4i}	0
5_i	4_i	1	0	0	0	$\pi/2$	0	q_{5i}	0

Dynamic modeling of the actual robot As proposed in (Briot and Gautier, 2012), the loop contraints can be taken into account by using the robot loop-closure equations (Merlet, 2006) and the Lagrange multipliers (Moon, 2007). The loop closure-equations of the Orthoglide can be expressed as

$$\mathbf{f}_t = \left[\mathbf{f}_{t_{11}}^T, \mathbf{f}_{t_{21}}^T, \mathbf{f}_{t_{12}}^T, \mathbf{f}_{t_{22}}^T, \mathbf{f}_{t_{13}}^T, \mathbf{f}_{t_{23}}^T\right]^T = \mathbf{0} \text{ with} \tag{2}$$

$$\mathbf{f}_{t_{1i}} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \mathbf{R}_0^i \begin{bmatrix} d_4 \cos q_{2i} \cos q_{3i} + a\delta_i \\ -d_4 \sin q_{3i} + a\delta_{i+1} \\ q_{1i} - a\bar{\delta}_{i+2} - d_4 \sin q_{2i} \cos q_{3i} + d_6 \end{bmatrix}, \mathbf{f}_{t_{2i}} = \begin{bmatrix} q_{2i} \\ q_{3i} \end{bmatrix} + \begin{bmatrix} q_{5i} \\ q_{4i} \end{bmatrix},$$

$$\mathbf{f}_p = \begin{bmatrix} x^2 + y^2 + (z - q_{11} - d_6)^2 - d_4^2 \\ (x - q_{12} - d_6 + a)^2 + y^2 + (z - a)^2 - d_4^2 \\ x^2 + (y - q_{13} - d_6 + a)^2 + (z - a)^2 - d_4^2 \end{bmatrix} = \mathbf{0}, \tag{3}$$

where $\mathbf{R}_0^i$ is the rotation matrix between the leg frame $(O_i, \mathbf{x}_{1i}, \mathbf{y}_{1i}, \mathbf{z}_{1i})$ and the base frame $(O_0, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ (Fig. 2c), d_6 is a length defined at Fig. 2b,

δ_i is a scalar of which value is equal to 1 if $i = 3$, 0 if not and $\bar{\delta}_i = 1$ ($\bar{\delta}_i = 0$, resp.) if $\delta_i = 0$ ($\delta_i = 1$, resp.).

As shown in (Briot and Gautier, 2012), the dynamic model of the real robot can be computed as:

$$\begin{aligned}\tau_{idm} &= \mathbf{J}_t^T \tau_{idm_t} + \mathbf{J}_p^T \mathbf{f}_p = \mathbf{J}_t^T \phi_{st_t}(\mathbf{q}_t, \dot{\mathbf{q}}_t, \ddot{\mathbf{q}}_t)\chi_{st_t} + \mathbf{J}_p^T \phi_p(\gamma)\chi_p \\ &= \begin{bmatrix}\mathbf{J}_t^T \phi_{st_t} & \mathbf{J}_p^T \phi_p\end{bmatrix} \begin{bmatrix}\chi_{st_t}^T & \chi_p^T\end{bmatrix}^T = \phi_{st}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\chi_{st}\end{aligned} \tag{4}$$

where χ_p represents the standard parameters of the platform (as the Orthoglide is a translational robot, the platform mass m_6 is the only identifiable parameter, i.e. $\chi_p = m_6$), $\gamma = [\ddot{x}, \ddot{y}, \ddot{z}]^T$ is the platform acceleration and $\phi_p = diag(\ddot{x}, \ddot{y} - g, \ddot{z})$. The same expressions can be used for adding in the model the contribution of the payload 7.

The matrices $\mathbf{J}_t$ and $\mathbf{J}_p$ and all positions, velocities and accelerations that are necessary for computing the dynamic model of the real robot can be straightforwardly calculated from (2) and (3) using expressions of (Briot and Gautier, 2012).

3 Total Least Square Identification (IDIM-TLS)

In this part, experimentations are performed and the dynamic model identification is carried out on the Orthoglide. The actuation of its feet is achieved by a rotary DC motor Sanyo Denki linked to a ball-screw. The actuators are powered by current source amplifiers Sanyo Denki. The global drive gains given by the manufacturer are equal to $637\pi \approx 2001$.

The approach presented in (Briot and Gautier, 2012) requires that the robot moves on two types of exciting trajectories: (i) a first type of trajectory without any payload and (ii) a second type of trajectory with a payload. The way to compute exciting trajectories for identification presented in (Gautier and Khalil, 1992) and is not detailed here. In our experiments, the payload mass has been measured with an accurate weighing machine ($M_7 = 1.983\text{kg} \pm 0.001\text{kg}$).

Table 3 presents the identification results. In this table, $2\sigma_{\hat{\chi}_i}$ represents the standard deviation of the parameters and $\%\sigma_{\hat{\chi}_{ri}}$ the percentage of estimation error.

Only the essential parameters are shown (Gautier, 1997), i.e. the parameters that have a real influence on the model. Let us recall that parameter g_{τ_j} represents the drive gain of actuator j, $XX_j, XY_j, XZ_j, YY_j, YZ_j, ZZ_j$ are the 6 components of the inertia matrix of link j at the origin of frame j, MX_j, MY_j, MZ_j are the 3 components of the first moment of link j, M_j is its mass, Ia_j is the total inertia moment for rotor and gears, Fv_j, Fc_j

are the visquous and Coulomb friction coefficients in the joint. Moreover, subscript 'R' stands for the parameters that have been regrouped (Gautier, 1991)[1]. It should also be noticed that the given value of standard deviation and percentage of error for the payload mass M_7 are those estimated by the use of the weighing machine.

The results show that the identified drive gains are about 5% far from those given by the manufacturer. With these identified parameters, the actuator efforts are reconstructed and compared with the measured ones (Fig 3). It can be seen that the efforts are well simulated.

In order to definitely validate our method, a second payload of 1.136 kg is mounted on the platform and a classical weighted least square identification is performed (Gautier, 1997). Two cases are considered: (Case 1) the drive gains used are those of the manufacturer and (Case 2) the drive gains are those identified in Table 3. The following results are found: (Case 1) $M_7 = 1.09$ kg with $2\sigma_{\hat{\chi}_i} = 0.02$ kg, (Case 2) $M_7 = 1.14$ kg with $2\sigma_{\hat{\chi}_i} = 0.02$ kg. It is thus possible to conclude that the identification of the drive gain using IDIM-TLS leads to better payload reconstruction.

4 Conclusions

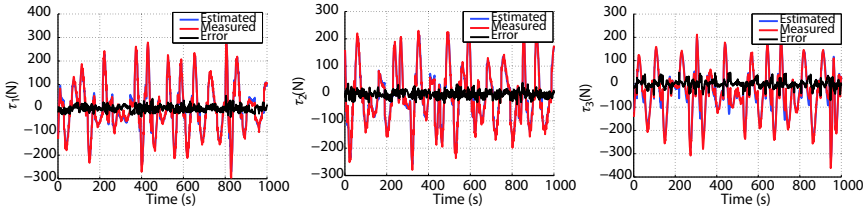
This paper has presented a global approach for both the identification of parallel robots dynamic parameters and drive gains. It is based on a IDIM-TLS technique using current reference and position sampled data while the

[1]

$$
\begin{aligned}
M_{11R} =& M_{11} + Ia_{11} + M_{21} + M_{31} + M_{41} + M_{51} + 3.23MX_{32} + M_{42} + M_{52} \\
&+ 3.23MX_{33} + M_{43} + M_{53} + M_6 \\
M_{12R} =& M_{21} + Ia_{21} + M_{22} - 3.23MX_{32} + M_{32} \\
M_{13R} =& M_{31} + M_{23} - 3.2258MX_{33} + M_{33} \\
ZZ_{21R} =& ZZ_{21} + Ia_{21} + YY_{31} + YY_{41} + 0.1M_{41} + 0.1M_{51} + 0.31MX_{32} + 0.1M_{42} \\
&+ 0.1M_{52} + 0.31MX_{33} + 0.1M_{43} + 0.1M_{53} + 0.1M_6 \\
Fv_{21R} =& Fv_{21} + Fv_{51} \\
ZZ_{31R} =& ZZ_{31} + Ia_{31} + 0.1M_{41} + 0.1M_{51} + 0.31MX_{32} + 0.1M_{42} + 0.1M_{52} \\
&+ 0.31MX_{33} + 0.1M_{43} + 0.1M_{53} + 0.1M_6 \\
MX_{31R} =& MX_{31} + 0.31M_{41} + 0.31M_{51} + MX_{32} + 0.31M_{42} + 0.31M_{52} + MX_{33} \\
&+ 0.31M_{43} + 0.31M_{53} + 0.31M_6 \\
Fs_{31R} =& Fs_{31} + Fs_{41} \\
Fs_{32R} =& Fs_{32} + Fs_{42} \\
Fv_{33R} =& Fv_{33} + Fv_{43} \\
Fs_{33R} =& Fs_{33} + Fs_{43}
\end{aligned}
$$

Table 3: Essential parameters of the Orthoglide.

Par.	Val.	$2\sigma_{\hat{\chi}_i}$	$\%\sigma_{\hat{\chi}_{ri}}$	Par.	Val.	$2\sigma_{\hat{\chi}_i}$	$\%\sigma_{\hat{\chi}_{ri}}$
g_{τ_1}	2110	8.93	0.42	Fs_{13}	24.4	0.0813	0.33
g_{τ_2}	2130	8.83	0.41	ZZ_{21R}	0.0970	$1.13 \cdot 10^{-3}$	1.17
g_{τ_3}	2070	5.16	0.25	MX_{21}	-0.123	$4.87 \cdot 10^{-3}$	3.95
M_{11R}	9.93	0.0468	0.47	Fv_{21R}	5.50	0.0796	1.45
Fv_{11}	83.0	0.377	0.45	ZZ_{31R}	0.0739	$1.04 \cdot 10^{-3}$	1.41
Fs_{11}	34.5	0.160	0.46	MX_{31R}	0.202	$3.43 \cdot 10^{-3}$	1.70
M_{12R}	8.86	0.0417	0.47	Fs_{31R}	1.60	0.0245	1.54
Fv_{12}	87.5	0.463	0.53	MX_{41}	0.0377	$7.73 \cdot 10^{-4}$	2.05
Fs_{12}	43.0	0.192	0.45	Fs_{32R}	1.57	0.0245	1.56
M_{13R}	1.10	0.0131	1.19	Fv_{33R}	-5.51	0.0851	1.54
Ia_{13}	7.93	0.0242	0.31	Fs_{33R}	1.37	0.0273	2.00
Fv_{13}	84.1	0.238	0.28	M_7	1.983	0.001	0.05



(a) Actuator 1 (b) Actuator 2 (c) Actuator 3

Figure 3: Measured and computed torques of the Orthoglide with the payload of 1.983 kg.

robot is tracking one reference trajectory without load fixed on the robot and one trajectory with a known payload fixed on the robot, whose inertial parameters are measured or calculated with a CAD software. Experiments show that the technique increase the identification quality and that with the new identified gains, the payload reconstruction is improved.

Bibliography

Y. Amiral, G.F. Francois, J. Pontnauand, and M. Dafaoui. Design and control of a new six-dof parallel robot: application to equestrian gait simulation. *Mechatronics*, 6:227239, 1996.

S. Briot and M. Gautier. Global identification of drive gains and dynamic parameters of parallel robots - part 1: Theory. In *Proceedings of the 19th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control (RoManSy)*, 2012.

D. Chablat, P. Wenger, and J.P. Merlet. A comparative study between two three-dof parallel kinematic machines using kinetostatic criteria and interval analysis. In *Proceedings 11th World Congress in Mechanism and Machine Science*, 2004.

M. Diaz-Rodriguez, V. Mata, A. Valera, and A. Page. A methodology for dynamic parameters identification of 3-dof parallel robots in terms of relevant parameters. *Mechanism and Machine Theory*, 45:13371356, 2010.

M. Gautier. Numerical calculation of the base inertial parameters. *Journal of Robotics Systems*, 8(4):485–506, 1991.

M. Gautier. Dynamic identification of robots with power model. In *Proceedings IEEE ICRA*, pages 1922–1927, Albuquerque, USA, April 1997.

M. Gautier and W. Khalil. Exciting trajectories for the identification of the inertial parameters of robots. *International Journal of Robotics Research*, 11(4):362–375, 1992.

M. Grotjahn, B. Heiman, and H. Abdellatif. Identification of friction and rigid-body dynamics of parallel kinematic structures for model-based control. *Multibody System Dynamics*, 11:273294, 2004.

S. Guegan, W. Khalil, and Ph. Lemoine. Identification of the dynamic parameters of the orthoglide. In *Proceedings IEEE ICRA*, pages 3272–3277, Taipei, Taiwan, September 2003.

M. Honegger, A. Codourey, and E. Burdet. Adaptive control of the hexaglide, a 6dof parallel manipulator. In *Proceedings of the IEEE ICRA*, page 543548, Albuquerque, NM, USA, 1997.

M. Honegger, R. Brega, and G. Schweitzer. Application of a nonlinear adaptive controller to a 6 dof parallel manipulator. In *Proceedings of the IEEE ICRA*, page 19301935, San Francisco, CA, USA, 2000.

W. Khalil and D. Creusot. Symoro+: a system for the symbolic modelling of robots. *Robotica*, 15:153–161, 1997.

W. Khalil and E. Dombre. *Modeling, Identification and Control of Robots.* Hermes Penton London, 2002.

W. Khalil, M. Gautier, and P. Lemoine. Identification of the payload inertial parameters of industrial manipulators. In *Proceedings IEEE ICRA*, pages 4943–4948, Roma, Italy, April 2007.

J.P. Merlet. *Parallel Robots.* Springer, 2nd edition, 2006.

F.C Moon. *Applied dynamics.* J. Wiley and Sons, 2007.

A. Vivas and P. Poignet. Predictive functional control of a parallel robot. *Control Engineering Practice*, 13:863874, 2005.

On the Conditioning of the Observation Matrix for Dynamic Parameters Identification of Parallel Robots

Miguel Díaz-Rodriguez[*], Vicente Mata[†], Angel Valera[‡] and Sebastian Provenzano[*]

[*] Departamento de Tecnología y Diseño, Facultad de Ingeniería, Universidad de los Andes, Mérida, Venezuela, dmiguel@ula.ve

[†] Centro de Investigación en Tecnología de Vehículos, Universidad Politécnica de Valencia, Spain, vmata@ula.ve

[‡] Departamento de Ingeniería y Sistemas y Automática, Universidad Politécnica de Valencia, Spain

Abstract Parallel robots can be modelled as serial robots after cutting-open the robot at some of its joints. Internal forces appear at joints which have been cut open. In this paper, the coordinate partitioning method and the velocity projection method are employed to make the internal forces disappear such that a dynamic parameter identification model can be obtained in terms of generalized independent coordinates. The influence of these methods for obtaining the identification model when exciting trajectories are used for experiments is investigated. In addition, the effect of the parameter identification solution is also considered.

1 Introduction

Dynamic parameter identification is a well-known technique for identifying the dynamic model of a robot. The identified model can be used either for developing model-based control schemes or for conducting realistic simulations of the robot, see Swevers et al. (1996) among others. In this technique, the dynamic model is fitted to measured data obtained experimentally when the robot follows a specific trajectory. On the whole, the rigid body model can be written in a linear form with respect to the dynamic parameters to be identified. Also, friction in joints can be included in the identification process if linear models with respect to the friction parameters are assumed. Thus, the identification problem is a Least Squares (LS) fit (or Weighted

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_13,

Least Squares, WLS). In the LS problem, an observation matrix which depends on the generalized coordinates and their time derivatives is related to the actuator forces through the dynamic parameters.

In the LS solution the error in the identification depends to a great extent on the conditioning of the observation matrix. Armstrong (1989) introduced the concept of "exciting trajectories" for the specific trajectories used in identification. The specific trajectories are designed through a nonlinear optimization process. Since then, several papers have focused on designing exciting trajectories, Gautier and Khalil (1992) among others for serial robots and Abdellatif and Heimann (2010); Díaz-Rodríguez et al. (2010b) for parallel robots. The widespread method considers trajectory points guided by a certain analytical function. Thus, finding the exciting trajectory consists of determining the parameters of this analytical function that satisfy any criterion.

Unlike serial robots, parallel robots have active and passive joints. Thus, to obtain the observation matrix, a method for projecting the generalized passive forces to the generalized active forces has to be used. Coordinate partitioning or the velocity projection method are two well-known techniques for dealing with this problem. In this paper, not only how those two procedures could affect the condition number of the observation matrix when a optimization procedure to find exciting trajectories is considered, but also their influence on the solution to the identification problem is studied.

This paper is organized as follows: Section II describes the development of the identification model for a parallel robot when using the coordinate partitioning or velocity projection methods. Section III presents the process of finding exciting trajectories. Section IV describes the case study: a 3-PRS parallel robot. The effect on parameter identification solution is also presented. The main conclusions are summarized in Section V.

2 Dynamic Model for a Parallel Robot

Parallel robots can be considered as a serial robot after cutting open the robot at some of its joints. Thus, internal generalized forces appear in a natural way in the dynamic formulation of the mechanical system. Since no experimental information is available for these forces, they must be eliminated from the dynamic formulation. This is an important drawback in dynamic parameter identification procedures. A variety of procedures, see Garcia de Jalon and Bayo (1994), have been proposed. These methods can be shared in two main groups: i) Coordinate partitioning and ii) Those that introduce an orthogonal complement of the Jacobian matrix.

When a mechanical system is modelled through a system of n dependent coordinates subject to m scleronomous constraints, the equation of motion, including the internal generalized forces, can be written as follows,

$$\mathrm{K} \cdot \vec{\Phi} = \vec{\tau} + (\mathrm{A}_q)^T \cdot \vec{\lambda} \tag{1}$$

where K is a matrix depending on the generalized coordinates and their time derivatives, A_q is the Jacobian matrix of the constraint and $\vec{\lambda}$ the vector of Lagrange multipliers. The vector $\vec{\Phi}$ contains the inertial terms for the moving links of the mechanical system considered, i.e.

$$\vec{\Phi} = \begin{bmatrix} m_i & mx_i & my_i & mz_i & Ixx_i & Ixy_i & Ixz_i & Iyy_i & Iyz_i & Izz_i \end{bmatrix}^T \tag{2}$$

for the $i-th$ link of the mechanism and $\vec{\tau}$ the vector of external generalized forces.

The general idea is to find a Matrix C, so that

$$\mathrm{A}_q \cdot \mathrm{C} = 0 \tag{3}$$

Coordinate partitioning

This method introduces an adequate partitioning into the dependent coordinate system used to model robot; so that, the equation of velocities could be written as follows,

$$\mathrm{A}_q^i \cdot \dot{\vec{q}}^i + \mathrm{A}_q^s \cdot \dot{\vec{q}}^s = 0 \tag{4}$$

where $\mathrm{A}_q^i, \mathrm{A}_q^s$ are the Jacobian matrix with respect to independent and secondary coordinates. From [4],

$$\dot{\vec{q}}^s = -\left(\mathrm{A}_q^s\right)^{-1} \cdot \mathrm{A}_q^i \cdot \dot{\vec{q}}^i \tag{5}$$

Considering [5] and the coordinate partitioning in [1], it is possible to write,

$$\left(\mathrm{K}^i - \mathrm{X}^T \cdot \mathrm{K}^s\right) \cdot \vec{\Phi} = \vec{\tau}^i - \mathrm{X}^T \cdot \vec{\tau}^s \tag{6}$$

where $\mathrm{X} = \left(\mathrm{A}_q^s\right)^{-1} \cdot \mathrm{A}_q^s$. After a suitable partitioning of the coordinates , the generalized external forces corresponding to the secondary coordinates will be null; therefore, expression 6 could be written as follows,

$$\mathrm{C}_1 \cdot \vec{\Phi} = \vec{\tau}^i \tag{7}$$

where $\mathrm{C}_1 = \left(-\mathrm{X}^T \cdot \mathrm{K}^s + \mathrm{K}^i\right)$

Velocity projection

In order to eliminate the term corresponding to internal forces in [1]. a projection method based on the QR decomposition (or SV decomposition, among others) of the Jacobian matrix can be considered, so that

$$\mathrm{A}_q = \mathrm{Q} \cdot \mathrm{R} \tag{8}$$

The orthogonal matrix Q could be expressed as,

$$\mathrm{Q} = \left[\begin{array}{cc} \mathrm{Q}^d & \mathrm{Q}^i \end{array} \right] \tag{9}$$

where the matrix Q^i with dimensions $N \times F$,F being the degrees of freedom of the system, has the well-known property of

$$\mathrm{A}_q \cdot \mathrm{Q}^i = 0 \tag{10}$$

thus, Q^i could be used to eliminate the internal forces from equation (10), in such a way that,

$$\mathrm{C}_2 \cdot \vec{\Phi} = \mathrm{Q}_i^T \cdot \vec{\tau} \tag{11}$$

where $\mathrm{C}_2 = \mathrm{Q}_i^T \cdot \mathrm{K}$

It must be mentioned that when the computational burden could be a problem, some procedures have been developed in order to avoid the QR decomposition for each new value of the generalized coordinates (see Garcia de Jalon and Bayo (1994)) using only one decomposition for all new configurations as possible. However, this is not the case in the parameter identification problem.

3 Exciting Trajectories

The dynamic model represented by equation [7] or [11] is a non-determined system in terms of the dynamic parameters. An overdetermined system could be obtained by applying the equation to different poses of the robot sampled through a prescribed trajectory. The result from the different poses, could be put together to form,

$$\mathrm{W}\left(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}\right) \cdot \vec{\Phi} = \vec{Y} \tag{12}$$

where W is called the observation matrix, and $\vec{Y}$ is the vector collecting the generalized forces of the different robot poses.

For a general mechanical system, the columns of W are not independent ones because some of the inertial parameters have no effect on the dynamics

of the system or affect in linear combinations. A set of base parameters has to be found such that,

$$\mathrm{W}_b\left(\vec{q},\dot{\vec{q}},\ddot{\vec{q}}\right)\cdot\vec{\Phi}_b=\vec{Y} \tag{13}$$

Equation [13] is now full-rank, thus the system that could be solved for the dynamic parameters either through the LS or WLS,

$$\Phi_{base}=\left(\mathrm{W}_b^T\,\Sigma^{-1}\,\mathrm{W}_b\right)^{-1}\,W\,\Sigma^{-1}\,\vec{\tau} \tag{14}$$

where Σ is a matrix of weights. The solution of equation [14] depends on the condition number of W matrix. Thus, the condition number of W is used in this paper as a criterion for finding the exciting trajectory. This can be established as,

$$\text{minimize cond}\left(\mathrm{W}_b\left(\vec{q},\vec{\dot{q}},\vec{\ddot{q}}\right)\right)\quad\text{subject to } g\left(\vec{q},\vec{\dot{q}},\vec{\ddot{q}}\right)\leq 0 \tag{15}$$

where $g\left(\vec{q},\vec{\dot{q}},\vec{\ddot{q}}\right)\leq 0$ are constraints regarding the actuator limits and workspace limitations. Equation [15] constitutes a nonlinear optimization problem.

For finding the exciting trajectory, the generalized coordinates are parametrized as Swevers et al. (1996) suggest,

$$q_i\left(t\right)=q_{i0}+\sum_{j=1}^{n_H}\left[\frac{a_{ij}}{2\pi f\cdot j}\sin\left(2\pi f\cdot j\cdot t\right)-\frac{b_{ij}}{2\pi f\cdot j}\cos\left(2\pi f\cdot j\cdot t\right)\right] \tag{16}$$

where t is the time, q_{i0}, a_{ij} and b_{ij} are the coefficients of the Fourier series that will be the variables of the optimization process, n_H is the harmonic number and f is the fundamental frequency.

4 Case Study

A 3-DOF parallel manipulator, see Díaz-Rodríguez et al. (2010b), was employed as a case of study. It consists of a base platform connected to a moving platform through three identical PRS limbs. The robot is actuated at the prismatic (P) joint by a DC motor connected to a ball screw which converts the rotational motion to translational.

4.1 Optimization process

Two sets of exciting trajectories were found. In one set the coordinate partitioning method is used to obtain W. The other set considers the veloc-

ity projection method. Hereinafter, the first set is called W1 and the second set W2. The number of design variables (np) of the optimization process depends on the number of harmonic functions $np = 3(2 \cdot nH + 1)$.Thus, nH should be kept as small as possible, since for higher values the optimization problem is more cumbersome. Another parameter that affects the optimization problem is the number of data samples N_{pts}. We have studied the effect of n_H and N_{pts} on the optimization process, which was carried out in a Matlab environment using the nonlinear constrained optimization routine ("fmincon").

The optimization process considers 10, 20, 30, 40, 50 and 100 data sampling points. In addition, the study was conducted considering 3 to 11 harmonic functions. From Figure 2 it can be noted that the observation matrix obtained by the projection method has a lower condition number than the one obtained by coordinate partitioning. At first glance it can be concluded that a better conditioning of the observation is obtained by the velocity projection method. It can also be seen that a number of harmonics as low as $nH = 7$ can be used to determine the exciting trajectories. Moreover, it can be seen that 60 data points are enough to get a good conditioning of the observation matrix.

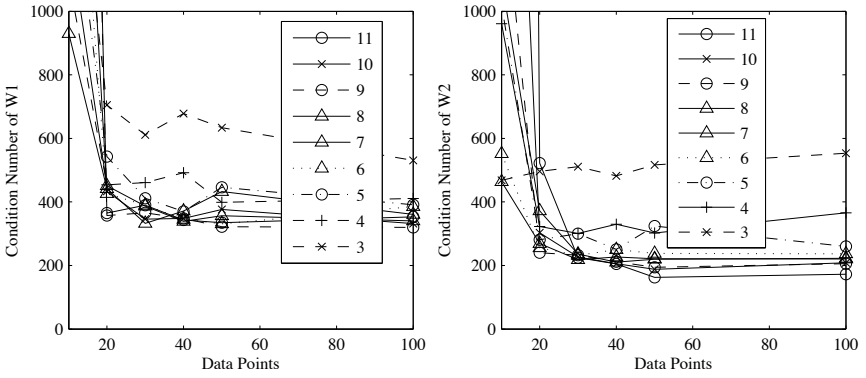


Figure 1. Condition number of W1 and W2 vs Data Points and n_H

4.2 Identification process

From previous results, 7 harmonic functions and 60 data points were employed in the optimization process. Eight trajectories were designed, four for each case (W1 and W2). The initial parameters for optimization are those obtained with harmonic functions with fewer parameters. In all

cases the condition number for W2 was lower than that of W1. Table 4.2 summarizes the mean and relative standard deviation values (σ) for the identified parameters. A methodology previously proposed by Díaz-Rodríguez et al. (2010a) was employed to find the relevant parameters. From the table it can be noted that the values for the identification are similar. The prediction forces error values, measured as the relative absolute error, obtained when using W1 for identification were lower than that obtained with W2.

Table 1. Dynamic Parameters

Φ_r W1	σ W1	Φ_r W2	σ W2
-20.41	1.22	-20.30	1.49
66.50	0.99	67.05	0.59
60.23	5.88	65.81	9.71
111.93	5.89	134.38	5.93
202.16	3.00	176.44	3.89
124.68	5.46	152.79	4.59
1917.78	2.12	1953.72	2.00
129.42	5.25	121.64	5.64
2091.01	1.96	2174.81	1.88
246.70	2.80	233.81	2.83
2072.81	1.94	2063.31	1.91
265.42	2.48	270.86	2.31

It was expected that the results from model W2 would be better that those of W1. However, in the velocity projection method not only passive forces are projected onto the independent generalized forces, but also actuated forces are projected onto this space, $Q_i^T \cdot \vec{\tau}$, see Figure 2. Coulomb friction forces from actuated joints are projected onto the independent generalized coordinates and, at some specific points these forces overlap, introducing error into the identification by LS or WLS.

5 Conclusion

Findings show that better conditioning could be obtained when using the velocity projection method and exciting trajectories are used for the dynamic parameter identifications of a parallel robot. However, when comparing the results from an identification point of view, no improvement is obtained. It is recommended to employ the coordinate partitioning method for building the observation matrix in the dynamic parameter identification of a parallel

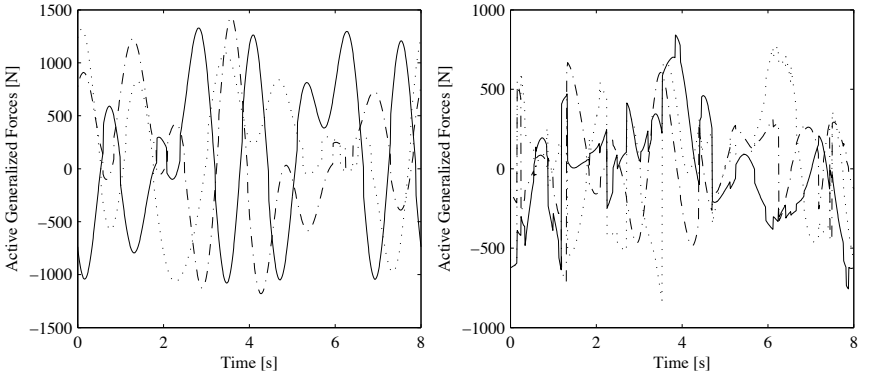


Figure 2. Active Generalized Forces for identification. Left $\vec{\tau}_i$ for W1 model and right $\mathrm{Q}_i^T \cdot \vec{\tau}$ for W2 model.

robot.

Bibliography

H. Abdellatif and B. Heimann. Experimental identification of the dynamics model for 6-dof parallel manipulators. *Robotica*, 28(3):359–368, 2010.

B. Armstrong. On finding exciting trajectories for identification experiments involving systems with nonlinear dynamics. *International Journal of Robotics Research*, 8(6):28–48, 1989.

M. Díaz-Rodríguez, V. Mata, A. Valera, and A. Page. A methodology for dynamic parameters identification of 3-dof parallel robots in terms of relevant parameters. *Mechanism and Machine Theory*, 45(9):1337 – 1356, 2010a.

M Díaz-Rodríguez, V. Mata, A. Valera, and A. Page. Forward dynamics of a 3-dof parallel robot: a comparison among different models. In *Proceedings of the Eighteenth CISM-IFToMM Symposium*, pages 160–165, 2010b.

J. Garcia de Jalon and E Bayo. *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge*. Springer-Verlag, 1994.

M. Gautier and W. Khalil. Exciting trajectories for the identification of base inertial parameters of robots. *International Journal of Robotics Research*, 11(4):362–375, 1992.

J. Swevers, C. Ganseman, J. DeSchutter, and H. VanBrussel. Experimental robot identification using optimised periodic trajectories. *Mechanical Systems and Signal Processing*, 10(5):561–577, 1996.

Optimal Force Generation of 3-RRR Decoupled Planar Robots for Ensuring Unlimited Platform Rotation

Sébastien Briot †, Vigen Arakelian ‡, Damien Chablat †, and Philippe Wenger †

† Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN) UMR CNRS 6592, Nantes, FRANCE

‡ Institut National des Sciences Appliquées de Rennes (INSA) M&CSE Department, Rennes, FRANCE

Abstract This paper deals with the optimal dynamics of the 3-RRR decoupled robot in which the linear displacements of the platform are decoupled from its orientation. The particularity of the 3-RRR decoupled robot is that an optimal choice of the leg's geometric parameters allows an unlimited platform rotation, which can be attractive for many industrial applications. However, a singular configuration is necessarily encountered during full platform rotation. In order to generate a stable motion in the presence of singularities, optimal dynamic conditions are disclosed. The suggested optimal conditions ensuring unlimited platform rotation are illustrated and validated by numerical simulations with ADAMS.

1 Introduction

In the last few decades, parallel manipulators have been the subject of various studies. Many problems devoted to their singularity, kinematics and dynamics properties were disclosed. Parallel manipulators are used in more and more applications nowadays. Consequently, new research problems are continuously appearing. One of the structural disadvantages of parallel manipulators is the low platform rotation capability due to the multi-branch architecture. In the last years this problem has been studied by several researchers. In (Liu et al., 2005) a new family of 3-DOF parallel manipulators with high rotational capability was proposed. The platform orientation range was increased by non symmetric placement of three non-identical legs. In (Sacludean et al., 2002; Company et al., 2005; Nabat et al., 2006b) new architectures of 3-DOF parallel manipulators with unlimited rotation of the end-effectors were developed. They consist of a pair of connected five-bar

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_14,

mechanisms. A similar approach was applied in a parallel robot for medical 3D-ultrasound imaging (Lessard et al., 2008, 2007). In the parallel manipulator H4, a complementary gear transmission (Kokikabushiki et al., 2000) together with a belt transmission (Nabat et al., 2006a) were used to amplify the orientation angle of the end-effector. A new family of spatial parallel mechanisms with decoupled and unlimited rotation of the moving platform was proposed in (Gogu, 2009). In the suggested architectures the moving platform performs two independent translations and one independent unlimited rotation whose axis is perpendicular to the plane of translation.

A modified mobile platform design (Fig. 1a) was proposed in a recent study (Arakelian et al., 2011), thus decoupling the position and the orientation of the platform. In the 3-RRR decoupled robot, the loop (O_1, A_1, C, A_2, O_2) moves point C and the leg (O_3, A_3, B_3) adjusts the orientation ϕ according to the position. If the position of point C is given, this third leg is equivalent to a four-bar linkage, which ensures a full rotation of the platform. For the generation of a full rotation of the platform, however, the four-bar linkage must cross a singularity.

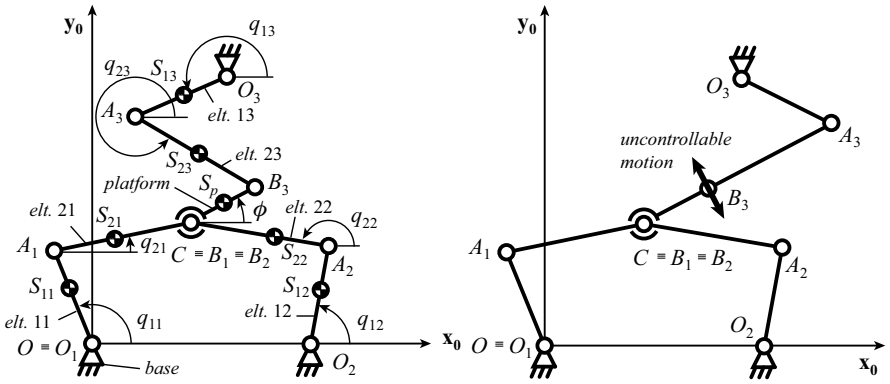
In this paper, the optimal dynamic conditions are found in order that the 3-RRR decoupled robot can cross a singularity during a full rotation motion of its platform.

2 Description of the 3-RRR Decoupled Robot

2.1 Structural features and singularities

Let us consider the architecture of the suggested manipulator (Fig. 1a). As shown in Fig. 1a, axis $\mathbf{x_0}$ is along vector $\overrightarrow{O_1O_2}$. The lengths of link O_iA_i (resp. A_iB_i) are denoted as L_{1i} (resp. L_{2i}). The length of the platform CB_3 is denoted as R. The positions of the base joints O_i along $\mathbf{x_0}$ and $\mathbf{y_0}$ are denoted as (x_{O_i}, y_{O_i}), with $x_{O_1} = y_{O_1} = y_{O_2} = 0$.

The aim of such a modification of the 3-RRR robot is to obtain an unlimited platform rotation. Indeed, the four-bar linkage, which generates the rotation of the platform, is equivalent to a rocker-crank mechanism. In this case, it is able to transfer the oscillation of the input link O_3A_3 to the full rotation of the platform CB_3. The robot should be designed so that this condition is satisfied in the whole workspace. However, a singularity must be encountered during a full platform rotation motion. This singularity occurs when CB_3 and A_3B_3 are aligned (Fig. 1b). As a result, the optimal generation of the input link O_3A_3 for passing through singular configurations must be achieved (Briot and Arakelian, 2008). This approach is based on the study of the robot dynamic model.



(a) Schematics (b) Example of singularity

Figure 1: Schematics of the 3-RRR decoupled parallel robot and its singular configurations encountered during a full platform rotation.

2.2 Dynamics of the robot

In this part, the robot inverse dynamic model is computed using the Lagrange equations. To simplify the computation, it is preferable to replace the masses of moving links $2i$ by concentrated masses (Seyferth, 1974). For a link ji with mass m_{ji} and its axial moment of inertia I_{ji}, we have:

$$\begin{bmatrix} 1 & 1 & 1 \\ r_{ji} & 0 & r_{ji}-1 \\ r_{ji}^2 L_{ji}^2 & 0 & (r_{ji}-1)^2 L_{ji}^2 \end{bmatrix} \begin{bmatrix} m_{ji1} \\ m_{ji2} \\ m_{ji3} \end{bmatrix} = \begin{bmatrix} m_{ji} \\ 0 \\ I_{ji} \end{bmatrix} \tag{1}$$

where:

- r_{ji} is the non-dimensional position of the centre of masses S_{ji} of element ji, i.e. for example for link 11, $OS_{11} = r_{11}L_{11}$,
- m_{jik} $(k = 1, 2, 3)$ are the values of the three point masses placed at the centers of the revolute joints and at the center of masses of the link ji.

The robot kinetic energy can be expressed as:

$$\begin{aligned} 2T &= \sum_{i,j} \left(m_{ji} \mathbf{v}_{S_j i}^2 + I_{ji} \dot{q}_{ji}^2 \right) + m_p \left(\dot{x}^2 + \dot{y}^2 \right) + I_p \dot{\phi}^2 \\ &= \sum_{i=1}^{3} \left(I_{r1i} \dot{q}_{1i}^2 + 2 I_{r2i} \dot{q}_{1i} (-\dot{x} \sin q_{1i} + \dot{y} \cos q_{1i}) \right) + m_{rp} \left(\dot{x}^2 + \dot{y}^2 \right) \\ &\quad + I_{rp} \dot{\phi}^2 + 2 I_{r3} \dot{q}_{13} \dot{\phi} \cos(q_{13} - \phi) + 2 I_{r4} \dot{\phi} (-\dot{x} \sin \phi + \dot{y} \cos \phi) \end{aligned} \tag{2}$$

where
$I_{r1i} = I_{1i} + (m_{1i}r_{1i}^2 + m_{2i1} + m_{2i2}(1 - r_{2i})^2)L_{1i}^2$,
$I_{r2i} = m_{2i2}r_{2i}(1 - r_{2i}))L_{1i}^2$,
$m_{rp} = m_p + m_{213} + m_{223} + m_{233} + m_{212}r_{21}^2 + m_{222}r_{22}^2 + m_{232}r_{23}^2$,
$I_{rp} = I_p + (m_{233} + m_{232}r_{23}^2)R^2$,
$I_{r3} = m_{232}r_{23}(1 - r_{23})RL_{13}$,
$I_{r4} = (m_p r_p + m_{233} + m_{232}r_{23}^2)R$,
and $\dot{x}$, $\dot{y}$ are the platform translational velocities along $\mathbf{x_0}$ and $\mathbf{y_0}$ axes and $\dot{\phi}$ is the platform rotation velocity, $\dot{q}_{1i}$ is the joint rate of actuator i, $\mathbf{v}_{S_{ji}}$ is the velocity of the centre of masses S_{ji}, $\dot{q}_{ji}$ the rotational velocity of link ji, m_p is the platform mass, I_p its axial moment of inertia expressed at C and r_p the non-dimensional position of the centre of masses S_p of the platform, i.e. $CS_p = r_p R$.

The gravity field is directed along the $\mathbf{z_0}$ axis. As a result, the dynamic model can be computed as (Moon, 2007):

$$\tau = \mathbf{w_b} - \mathbf{B}^T\lambda, \text{with } \mathbf{A}^T\lambda = \mathbf{w_p} \tag{3}$$

where λ are the Lagrange multipliers, matrices $\mathbf{A}$ and $\mathbf{B}$ are defined such that $\mathbf{At} + \mathbf{B\dot{q}} = \mathbf{0}$ for $\mathbf{t} = [\dot{x}, \dot{y}, \dot{\phi}]^T$ and $\dot{\mathbf{q}} = [\dot{q}_{11}, \dot{q}_{12}, \dot{q}_{13}]^T$ and

$$\mathbf{w_b} = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial T}{\partial \mathbf{q}}, \mathbf{w_p} = \frac{d}{dt}\left(\frac{\partial T}{\partial \mathbf{t}}\right) - \frac{\partial T}{\partial \mathbf{x}}, \mathbf{x} = [x, y, \phi]^T \tag{4}$$

3 Optimal Motion Generation

3.1 Condition for crossing the singularities

As disclosed in (Briot and Arakelian, 2008), the singularities considered in this work can be crossed if and only if:

$$\mathbf{t}_s^T \mathbf{w_p} = 0, \tag{5}$$

where $\mathbf{t}_s$ is the unit screw describing the gained degree of freedom when the robot encounters a singularity. For the robot at hand, for the 'orientation' singularity (Fig. 1b), it can be proven that $\mathbf{t}_s$ is defined by $\mathbf{t}_s = [0, 0, 1]^T$ throughout the workspace. Therefore, the condition for crossing this type of singularity remains to cancel the last line of $\mathbf{w_p}$, i.e. from (2) and (4)

$$I_{rp}\ddot{\phi} + I_{r3}(\ddot{q}_{13}\cos(q_{13} - \phi) - \dot{q}_{13}^2\sin(q_{13} - \phi)) + I_{r3}(\ddot{y}\cos\phi - \ddot{x}\sin\phi) = 0, \tag{6}$$

To lower the computation burden, it is preferable to express equation (6) as a function of $\dot{x}$, $\dot{y}$, $\dot{\phi}$, $\ddot{x}$, $\ddot{y}$ and $\ddot{\phi}$ only. The terms $\dot{q}_{13}$ and $\ddot{q}_{13}$ can be

removed using the following kinematic relationships:

$$\begin{aligned}\dot{\mathbf{q}} &= -\mathbf{B}^{-1}\mathbf{A}\mathbf{t}\\ \ddot{\mathbf{q}} &= -\mathbf{B}^{-1}(\mathbf{A}\dot{\mathbf{t}} + \dot{\mathbf{A}}\mathbf{t} + \dot{\mathbf{B}}\dot{\mathbf{q}}) = -\mathbf{B}^{-1}(\mathbf{A}\dot{\mathbf{t}} + (\dot{\mathbf{A}} - \dot{\mathbf{B}}\mathbf{B}^{-1}\mathbf{A})\mathbf{t})\end{aligned} \tag{7}$$

The values of $\dot{q}_{13}$ and $\ddot{q}_{13}$ can be extracted from the last line of vectors $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$.

3.2 Illustrative example

In order to validate the previous approach, an ADAMS model is developed with the following robot parameters extracted from (Chablat et al., 2012): $R = 0.1$ m; $L_{1i} = L_{2i} = 0.33$ m; $x_{O_2} = 0.1$ m; $x_{O_3} = 0.05$ m; $y_{O_3} = 0.6$ m; $m_{1i} = m_{2i} = 1$ kg; $m_p = 0.5$ kg; $I_{1i} = I_{2i} = 0.009$ kg.m^2; $I_p = 0.0017$ kg.m^2; $r_{ji} = r_p = 0.5$.

The robot is located at $\mathbf{x}(t_0) = \mathbf{x_{in}} = [-0.1\text{ m}, 0.45\text{ m}, 90\text{ deg}]^T$ for $t_0 = 0$ s. Its task is to maintain the position of point C while achieving a full rotation of the platform in $t_f = 1.5$ s. Singularities will appear for $\phi_{s1} \approx 269$ deg and $\phi_{s2} \approx 328$ deg. Two cases will be treated: (i) Case 1: a motion without taking into account the conditions for passing through the singularities, and (ii) Case 2: a motion that considers the conditions for passing through the singularities.

For Case 1, the motion planning is achieved with a 7th degree polynomial law in order to cancel the velocity, acceleration and jerk at the beginning and at the end of the trajectory:

$$\phi(t) = 360s(t)+90,\ s(t) = 35(t/t_f)^4 - 84(t/t_f)^5 + 70(t/t_f)^6 - 20(t/t_f)^7 \tag{8}$$

where the following initial and final conditions are respected:
$s(t_0) = \dot{s}(t_0) = \dot{s}(t_f) = \ddot{s}(t_0) = \ddot{s}(t_f) = \dddot{s}(t_0) = \dddot{s}(t_f) = 0, s(t_f) = 1$.

The input torque of actuator 3 is plotted in Fig. 2a (it can be proven that if the platform position is not changed, the torques of actuators 1 and 2 are cancelled). It is apparent that the input torque tends to infinity at each singularity.

For Case 2, the motion is planned using three different polynomials s_k. Each part of the trajectory has to be achieved in 0.5s. s_1 is defined for $\phi \in [90\text{deg}, \phi_{s1}]$ on the time interval $[t_0, t_{f1} = 0.5\text{s}]$, s_2 for $\phi \in [\phi_{s1}, \phi_{s2}]$ on the time interval $[t_{f1}, t_{f2} = 1\text{s}]$, and s_3 for $\phi \in [\phi_{s2}, 450\text{deg}]$ on the time interval $[t_{f2}, t_{f3} = 1.5\text{s}]$:

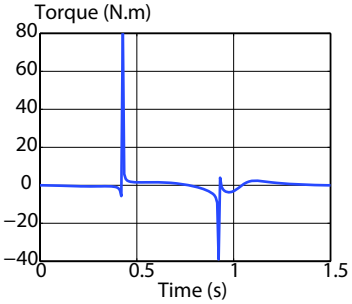
$$\begin{aligned}\phi(t) &= (\phi_{s1} - 90)s_1(t) + 90,\ \text{for } t \in [t_0, t_{f1}]\\ \phi(t) &= (\phi_{s2} - \phi_{s1})s_2(t) + \phi_{s1},\ \text{for } t \in [t_{f1}, t_{f2}]\\ \phi(t) &= (450 - \phi_{s2})s_3(t) + \phi_{s2},\ \text{for } t \in [t_{f2}, t_{f3}]\end{aligned} \tag{9}$$

with

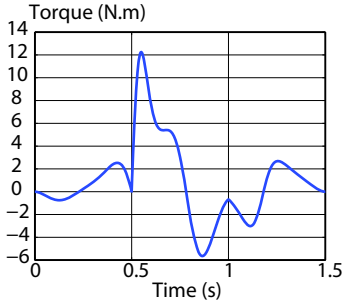
$$
\begin{aligned}
s_1(t) =& 441.8819t^4 - 2074.5386t^5 + 3411.5717t^6 - 1932.0441t^7,\\
s_2(t) =& 0.1690t + 0.0159(t-t_{f1})^2 + 443.8291(t-t_{f1})^4 - 2116.1982(t-t_{f1})^5\\
&+ 3521.1306(t-t_{f1})^6 - 2011.4236(t-t_{f1})^7,\\
s_3(t) =& 2.4400(t-t_{f2}) + 3.8429(t-t_{f2})^2 + 15.8838(t-t_{f2})^4\\
&- 316.3353(t-t_{f2})^5 + 746.8231(t-t_{f2})^6 - 506.5084(t-t_{f2})^7,
\end{aligned}
$$

where the following initial and final conditions are respected in order to ensure the velocity, acceleration and jerk continuity all along the trajectory:

$$
\begin{aligned}
& s_1(t_0) = \dot{s}_1(t_0) = \ddot{s}_1(t_0) = \dddot{s}_1(t_0) = \dddot{s}_1(t_{f1}) = 0, s_1(t_{f1}) = \dot{s}_1(t_{f1}) = 1,\\
& \ddot{s}_1(t_{f1}) = 0.0318,\\
& s_2(t_{f1}) = \dddot{s}_2(t_{f1}) = \dddot{s}_2(t_{f2}) = 0, s_2(t_{f2}) = \dot{s}_2(t_{f2}) = 1, \ddot{s}_2(t_{f2}) = 7.6858,\\
& \dot{s}_2(t_{f1}) = \dot{s}_1(t_{f1})d_1, \ddot{s}_2(t_{f1}) = \ddot{s}_1(t_{f1})d_1, d_1 = (\phi_{s1} - 90)/(\phi_{s2} - \phi_{s1})\\
& s_3(t_{f2}) = \dot{s}_3(t_{f3}) = \ddot{s}_3(t_{f3}) = \dddot{s}_3(t_{f2}) = \dddot{s}_3(t_{f3}) = 0, s_3(t_{f3}) = 1,\\
& \dot{s}_3(t_{f2}) = \dot{s}_2(t_{f2})d_2, \ddot{s}_3(t_{f2}) = \ddot{s}_2(t_{f2})d_2, d_2 = (\phi_{s2} - \phi_{s1})/(450 - \phi_{s2}).
\end{aligned}
$$



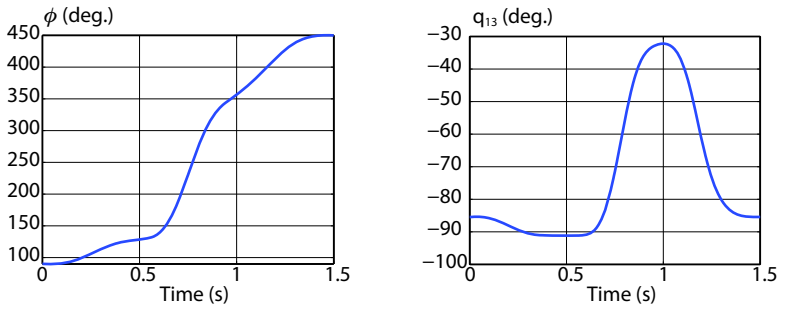
(a) For the trajectory defined at (8)



(b) For the trajectory defined at (9)

Figure 2: Torque of actuator 3.

It should be mentioned that the value of $\ddot{s}_1(t_{f1})$ ($\ddot{s}_2(t_{f2})$, resp.) is calculated so that it respects the condition for crossing the singularities expressed at (6), taking into account that $\dot{s}_1(t_{f1}) = 1$ ($\dot{s}_2(t_{f2}) = 1$, resp.). The value of ϕ all along this trajectory is plotted at Fig. 3a. Figure 3b presents the resulting displacement of actuator 3. It can be observed that this actuator has an oscillatory behaviour while the platform is ensuring a full rotation.



(a) Value of the platform rotation ϕ (b) Resulting displacement of actuator 3

Figure 3: The trajectory defined at (9).

The input torque of actuator 3 is plotted in Fig. 2b. It appears now that when crossing the singularities, the input torques remain finite.

It should be noted that in this example, only one turn has been achieved with a fixed position of point C, but it is obvious that the number of platform turns may be increased and that point C can move as soon as the global trajectory respects equation (6) at each singular configuration.

4 Conclusions

It is known that the singularity leads to a loss of robot controllability and it is a serious obstacle for reproduction of stable motions. This drawback can be overcome by means of the optimum force control. In this paper, the optimal force generation of 3-RRR decoupled robots for ensuring unlimited platform rotation has been studied. The particularity of these robots is that an optimal choice of the legs' geometric parameters allows an unlimited platform rotation. However, a singular configuration is necessarily encountered during a full platform rotation. In order to generate a stable motion in the presence of singularities, optimal dynamic conditions were found. To illustrate the fruitfulness of the obtained results two different types of control have been compared: a motion generated via a 7th degree polynomial law without taking into account the conditions for passing through the singularity and a motion that considers the optimal dynamic conditions for passing through the singularity. For the second case, a full rotation of the platform without perturbation of motion has been achieved. The obtained results showed that the generation of stable unlimited platform rotation of 3-RRR decoupled robots is possible. It can be attractive for many industrial

applications.

Bibliography

V. Arakelian, S. Briot, S. Yatsun, and A. Yatsun. A new 3-dof planar parallel manipulator with unlimited rotation capability. In *Proc. 13th World Congress in Robot and Machine Science*, 2011.

S. Briot and V. Arakelian. Optimal force generation of parallel manipulators for passing through the singular positions. *International Journal of Robotics Research*, 27:967–983, 2008.

D. Chablat, G. Moroz, V. Arakelian, S. Briot, and P. Wenger. Solution regions in the parameter space of a 3-RRR decoupled robot for a prescribed workspace. In *Proc. Advances in Robot Kinematics, submitted to*, 2012.

O. Company, F. Pierrot, V. Nabat, and M. Rodriguez. Schoenflies motion generator: A new non redundant parallel manipulator with unlimited rotation capability. In *Proceedings ICRA 2005*, pages 3250–3255, 2005.

G. Gogu. Parallel mechanisms with decoupled rotation of the moving platform in planar motion. *Journal of Mechanical Engineering Sciences, Part C*, 224:709–720, 2009.

K.T. Kokikabushiki, F. Pierrot, and O. Company. 4-dof parallel robot, patent, 2000.

S. Lessard, P. Bigras, and I.A. Bonev. A new medical parallel robot and its static balancing optimization. *Journal of Medical Devices*, 1:272–278, 2007.

S. Lessard, I.A. Bonev, and Patent P. Bigras. Parallel manipulator, 2008.

X.-J. Liu, J. Wang, and G. Pritschow. A new family of spatial 3-dof fully-parallel manipulators with high rotational capability. *Mechanism and Machine Theory*, 40:475–494, 2005.

F.C Moon. *Applied dynamics*. J. Wiley and Sons, 2007.

V. Nabat, F. Pierrot, M.R. Mijangos, J.M.A. Arteche, R.B. Zabalo, O. Company, and K.F. Perez De Armentia. High-speed parallel robot with four degrees of freedom, patent, 2006a.

V. Nabat, F. Pierrot, M. Rodriguez, J.M. Azcoita, R. Bueno, O. Company, and K. Florentino. Unlimited-rotation parallel robot with four degrees of freedom, patent, 2006b.

S.E. Sacludean, L.J. Stocco, and I.C.N. Chau. 3-dof parallel manipulator, patent, 2002.

W. Seyferth. Massenersatz duch punktmassen in rumlichen getrieben. *Mechanism and Machine Theory*, 9:49–59, 1974.

A New Algorithm for Gravity Compensation of a 3-UPU Parallel Manipulator

M. Banitalebi Dehkordi , A. Frisoli , E. Sotgiu and M. Bergamasco *
PERCRO, TeCIP Institute, Scuola Superiore Sant'Anna

Abstract This paper presents a new approach for gravity compensation of a 3-UPU parallel manipulator. The conventional method of computing the effect of gravity force on the end-effector works properly for serial manipulators. However, employing Newton-Euler approach for the parallel robots is computationally expensive and it cannot satisfy the requirements in this work. In order to overcome this difficulty, the new algorithm based on Lagrengian method is proposed. This model is established based on total potential energy of the system as a scalar value and, the position of the end-effector. This paper presents this *new algorithm* which is more efficient in sense of computation and more proper for Real-Time purposes in parallel robots.

1 Introduction

In the recent years, parallel robots have attracted the attention of many researchers worldwide. Literatures from Han et al. (2002), Tsai and Joshi (2000), Li and Xu (2006) show that parallel robot is preferred over its serial counterpart, because of presenting high load-carrying capacity, high velocity, structural stiffness, precision and low inertia. The main drawbacks of parallel robots are complexities of mechanical designs, motion generation, direct kinematics and control, and above all, the limited workspace. As an alternative, many researchers design different parallel manipulators with different degrees of freedom. Some examples are the famous translational 3 DOF Delta robot (Siciliano and Khatib, 2008, part. B/12), the SNU 3-UPU mechanism (Walter et al., 2008), Stewart platform and several other 3 DOF and 6 DOF parallel mechanisms (Wang and Xi, 2008).

For several decades, gravity compensation has been of great interest to researchers. Gravity compensation is considered as mechanical compensation, which is to design the mechanical components in order to satisfy

*This research was carried out with the financial support of BRAVO and VERE projects.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_15,

statically balancing of the robot, and active compensation through actuators. The literatures of Wang and Xi (2008) and Siciliano and Khatib (2008) show that in the effort of mechanical gravity compensation for serial mechanisms mainly springs and counterweights and in a few cases pulley and cams are employed, while for their parallel counterparts usually different types of springs and sometimes counterweights are used (Deepak and Ananthasuresh, 2011, and Siciliano and Khatib, 2008). In the absence of mechanical gravity compensation components, actuators should provides sufficient torques in order to compensate the force imposed by mechanism weight on the end-effector.

The gravity compensation of 3-UPU parallel manipulator is of high importance, while its translational movement makes it a suitable robot to be used widely in industry. The objective of this work is the development and experimental evaluation of a computationally efficient algorithm, for on-line gravity compensation of a 3-UPU parallel manipulator.

The remainder of this paper is organized as follows. Section 2 covers a brief description of fabricated parallel manipulator. Section 3 describes the proposed gravity compensation algorithm and finally, the experimental results discussed in section 4.

2 Description of device

The main objective of this project is to develop an fMRI compatible haptic interface, in order to utilize in neuroscience field. Employing this device and simulating the human daily life interactions and environment, the brain activity would be the most similar to the one during Activities of Daily Living (ADLs). In addition, the valuable information for studying the brain areas that participate in motor control can be provided.

To satisfy the project requirements a 3-UPU parallel manipulator is designed and fabricated, as shown in Fig. 1.a. Considering a strong magnetic field in the fmri environment, the device is fabricated using magnetic compatible materials such as carbon fiber, polymer and non-ferromagnetic metals such as aluminium and brass. Three shielded DC motors are used as links actuators. In order to avoid affecting the brain image and secure safety, the distance between fMRI scanner and shielded motors should be more than 1.8 m (Li et al., 2010).

Fig. 1.b illustrates a schematic model and relevant coordinates of 3-UPU parallel robot, which consists of a fixed base (an isosceles triangle) and a moving platform (an equilateral triangle). These two planes are connecting through three identical links, which are jointed to the planes by universal joints. At the orthocentre of each triangular platform, a Cartesian coordi-

nates is attached. The end-effector is fixed to point B with a small offset. On account of the pure translational motion of device, geometric method is used to obtain the workspace of manipulator (Siciliano and Khatib, 2008). Fig. 1.c shows the theoretical workspace of the device, which is the intersection of three spheres with radius of maximum link length. In practice, due to the mechanical constraints manipulator's workspace becomes smaller as shown in Fig. 1.d.

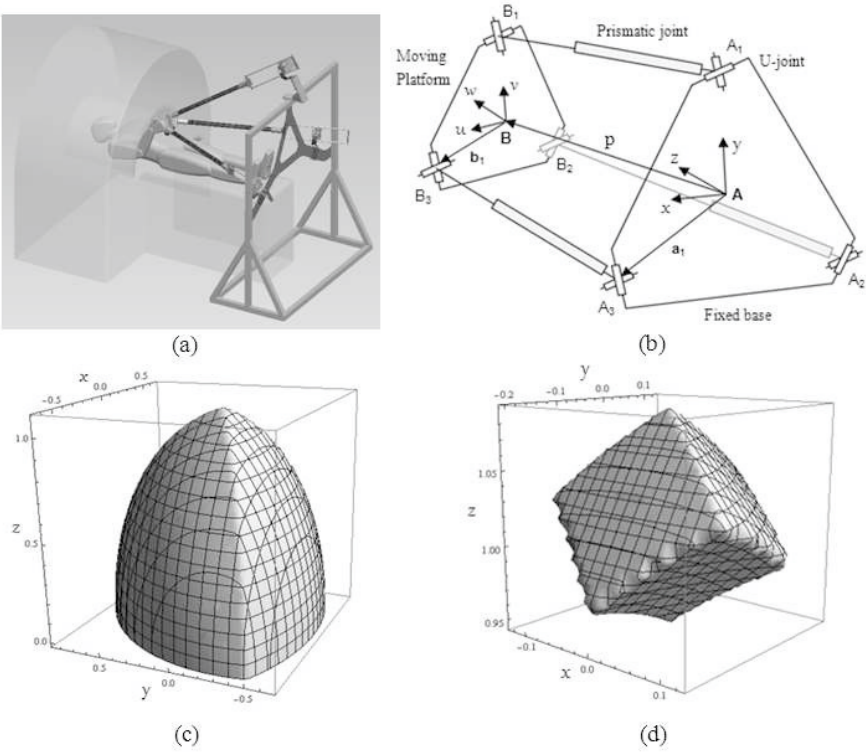


Figure 1. 3-UPU Parallel Manipulator. **a**. The model of device and fMRI scanner. **b**. A schematic diagram. **c**. Theoretical Workspace. **d**. Reachable Workspace.

3 Gravity compensation

Static balancing of parallel manipulator is an important issue and of great interest especially in the presence of heavy links in the mechanism. The conventional Newton-Euler method for gravity compensation in parallel robots leads to complexity and limitation in obtaining solutions and in this way, it has limited the feasible designs. Therefore a new method is needed in order to improve the practicality of balancing (Deepak and Ananthasuresh, 2011, and Siciliano and Khatib, 2008). In this proposed method, in order to obtain feasible solutions, the force imposed by links weight on the end-effector is calculated getting use of total potential energy of the system.

As Gosselin discussed (Wang and Xi, 2008, chap. 2) the position vector of the centre of mass of manipulator can be written as

$$\mathbf{c} = \frac{1}{M} \sum_{i=1}^{n} m_i \mathbf{c}_i \tag{1}$$

where n is a number of moving bodies in the mechanism, m_i is the mass of ith moving body and M is the total mass of moving bodies.

In the presence of elastic elements, the condition for static balancing is that the total potential energy of the system, which is the sum of gravitational and elastic potential energy, is constant (Wang and Xi, 2008).In the absence of spring and other energy storing elements, the condition for statically balancing of the mechanism can be written as

$$\mathbf{e}_z^T \mathbf{c} = C_t \tag{2}$$

where $\mathbf{e}_z$ is a unit vector representing the direction of gravity and C_t can be any arbitrary constant. In general, the position vector of centre of mass is depending on the manipulator configuration.

In order to find the centre of mass of 3-UPU parallel manipulator, assume P is a position of the end-effector in the Cartesian coordinate, which its origin is attached to the fixed base at point A as illustrated in Fig. 1.b. For this mechanism 7 moving bodies are considered, 6 refer to carbon beams of the links and aluminium bars that formed actuators, and one for the end-effector. In this manipulator $\mathbf{e}_z$ is along y-axis of defined Cartesian coordinate. Solving *Direct Kinematics* and *Inverse Kinematics*, both links' length $\mathbf{d}$ and position of the end-effector $\mathbf{P}$ are known in any configuration (Tsai and Joshi, 2000). By projecting the end-effector and three links on the fixed base, position of centres of mass along y-axis can be obtained as

$$C_{ci,Ali} = P_y - L_{gi,Ali} \sin\varphi_i \sin\psi_i \qquad i = 1, 2, 3 \tag{3}$$

where L_{gi} and L_{Ali} are constant based on structural properties. φ_i is an angle between x-axis and projection of the ith link on the fixed base, P_y is the position of end-effector along y-axis, ψ_i is an angle between the ith link and z-axis and C_{ci} and C_{Ali} are the position of centre of mass of the carbon beam and aluminium bars of the ith link along y-axis.

Therefore, the total potential energy of the system can be written as

$$U = g\mathbf{e}_z^T \sum_{i=1}^{7} m_i \mathbf{c}_i \tag{4}$$

Without loss of generality, one can obtain

$$\mathbf{u} = [\mathbf{M}_{ee} \;\; \mathbf{M}_c \;\; \mathbf{M}_{Al}] \begin{bmatrix} P_y & \cdots & 0 \\ \vdots & C'_c & \vdots \\ 0 & \cdots & C'_{Al} \end{bmatrix} \tag{5}$$

where $\mathbf{M}_{ee}$ is a matrix of size 3×1 and $\mathbf{M}_i$ and $\mathbf{M}_{Ali}$ are matrices of size 3×3, which are masses of end-effector, carbon beams i and aluminium bars of ith prismatic joint respectively, for i=1,2,3. C'_c and C'_{Al} are matrices of size 3×3 which are the centre of mass position of the carbon beams and aluminium bars respectively.

Following this notation, total potential energy can be written as

$$U = g\mathbf{e}_z^T \mathbf{u}\kappa \tag{6}$$

where κ is an arbitrary constant vector of size 7×1.

From Equ. 6 the potential energy can be calculated as a function of position of the end-effector $\mathbf{P}$. The force imposed by mechanism weight at the end-effector F_g can be obtained by taking derivation of U respect to $\mathbf{P}$. Therefore, the condition of gravity compensation can be written as

$$F_g + F_m = C_f \tag{7}$$

where F_g and F_m are gravitational force and motors force respectively, and C_f is an arbitrary constant, i.e. $C_f = 0$ in a complete compensation.

Deriving from (Checcacci et al., 2002), the gravitational force can be obtained as

$$\mathbf{F} = \lim_{\Delta P \longrightarrow 0} \frac{\Delta U(\mathbf{P})}{\Delta \mathbf{P}} = \frac{\partial U(\mathbf{P})}{\partial \mathbf{P}} \tag{8}$$

From Equ. 6 and 8 gravitational force at the end effector in the Cartesian space can be obtained as

$$\Lambda = \frac{\partial U}{\partial \varphi} = \begin{bmatrix} \frac{\alpha_1 \cos\varphi_1 (P_y - \alpha_2)}{\sqrt{{P_x}^2 + (P_y - \alpha_2)^2}} \\ \frac{\alpha_1 \cos\varphi_2 (P_y + \alpha_3)}{{P_z}^2 + \sqrt{(P_x + \alpha_4)^2 + (P_y + \alpha_5)^2 + {P_z}^2}} \\ \frac{\alpha_1 \cos\varphi_3 (P_y + \alpha_3)}{{P_z}^2 + \sqrt{(P_x - \alpha_4)^2 + (P_y + \alpha_5)^2 + {P_z}^2}} \end{bmatrix}^T \tag{9}$$

where α_is are constants which are conditional upon device structural properties.

$$\Xi = \frac{\partial \varphi}{\partial \mathbf{P}} = \begin{bmatrix} \frac{P_z P_x}{{B_1}^{3/2}\left(1 - \frac{{P_z}^2}{B_1}\right)^{1/2}} & \cdots & 0 \\ \vdots & \frac{\beta_1 P_z (\beta_2 P_x + \beta_3)}{{B_2}^{3/2}\left[(1 - \frac{{P_z}^2}{B_2}\right)^{1/2}} & \vdots \\ 0 & \cdots & \frac{\beta_1 P_z (\beta_2 P_x - \beta_3)}{{B_3}^{3/2}\left(1 - \frac{{P_z}^2}{B_3}\right)^{1/2}} \end{bmatrix} \tag{10}$$

where

$$B_i = (P_x + \gamma_{1i})^2 + (P_y + \gamma_{2i})^2 + {P_z}^2 \tag{11}$$

and β_is and $\gamma_{1,2i}$s are constants which their values are obtained based on device structural properties. From Equ. 9 and 10, $\mathbf{F}$ can be obtained in term of $\mathbf{P}$ as follow

$$\mathbf{F} = \Lambda\Xi = \frac{\partial U}{\partial \varphi}\frac{\partial \varphi}{\partial \mathbf{P}} = \left[\; F_x(P_x, P_y, P_z), \;\; F_y(P_x, P_y, P_z), \;\; F_z(P_x, P_y, P_z) \; \right] \tag{12}$$

where F_x, F_y and F_z are components of F in the Cartesian space, and are functions of the position of the end-effector.

4 Experiments

In order to demonstrate the reliability of the model some experimental tests were carried out. Employing force sensor to the end-effector, the gravitational force imposed to the end-effector was measured. The sensitivity and rated capacity of the force sensor were $10,015[V]$ and $245,166[N]$ respectively. End-effetor was placed in different trial positions in the workspace and achieved data from sensor and model were saved separately. In order to compare data from sensor with the one from model, for any configuration, the mean value of recived data over the 1 second of time was calculated. Fig. 2.a shows the fabricated manipulator and the attached force sensor to the end-effector. Fig. 2.b shows the trial positions of the end-effector during the test. Fig. 2.c illustrates the comparison between values of force

obtained from model and sensor and the error, which evaluated as a difference of forces over force from model. Finally, the value of actuators currents relevant to each trial position is shown in Fig. 2.d. The existing error is due to the presence of friction in the robot joints, which was not compensated by this algorithm. The experimental results were satisfactory while the maximum error is 18% and gravity compensation implemented successfully for the whole workspace, however the friction compensation will be taken into account in the future work.

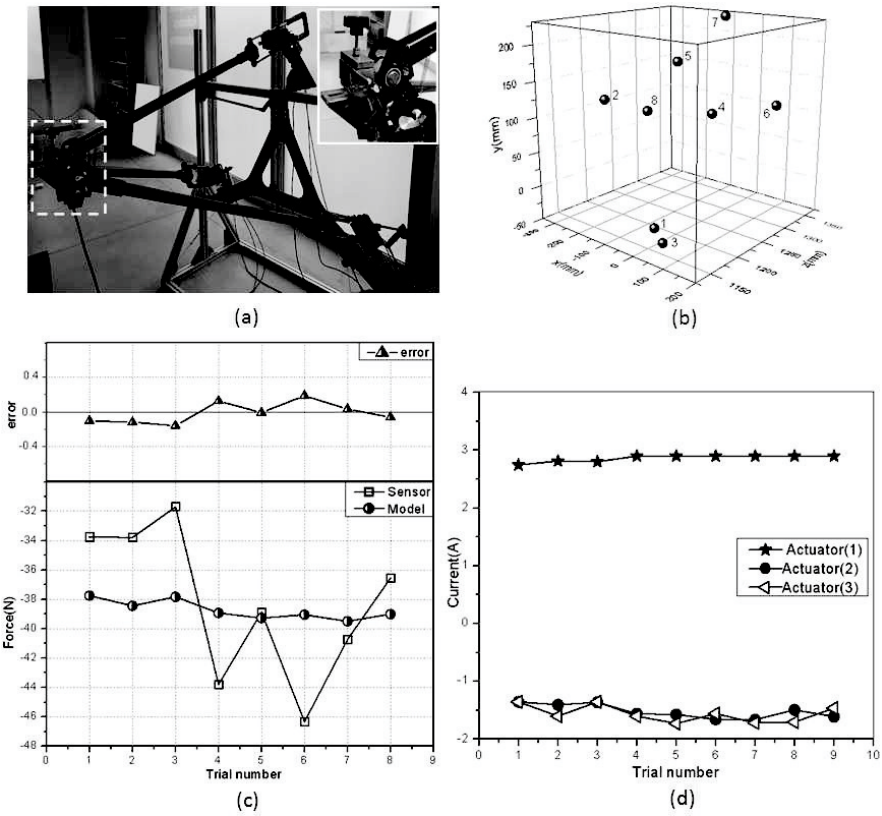


Figure 2. Experimental results. **a**. 3-UPU parallel manipulator during the experiment. **b**. Trial positions of end-effector with numbering. **c**. Comparison of forces obtained from model and sensor, and relative error. **d**. The relevant actuators currents.

5 Conclusion

Nowadays parallel robots are finding their path into several applications in industry and science, but their analysis is still far from being complete compare to their serial counterparts. Some of conventional methods that work properly for serial robots, show some weakness employing in the parallel manipulators. This paper demonstrates the new algorithm based on Lagrangian method, which gets advantage of potential energy. This model is successfully implemented for the 3-UPU parallel manipulator, and experiments illustrate the feasibility of the proposed algorithm to any parallel manipulator with the same structure, knowing the total potential energy of the system and the position of the end-effector. For the future work, we will supplement the mechanical gravity compensation to reduce the burden of actuators.

Bibliography

D. Checcacci, E. Sotgiu, A. Frisoli, CA Avizzano, and M. Bergamasco. Gravity compensation algorithms for parallel haptic interface. In *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*, pages 140–145. IEEE, 2002.

S.R. Deepak and GK Ananthasuresh. Static balancing of a four-bar linkage and its cognates. *Mechanism and Machine Theory*, 2011.

C. Han, J. Kim, J. Kim, and F.C. Park. Kinematic sensitivity analysis of the 3-upu parallel mechanism. *Mechanism and Machine Theory*, 37(8): 787–798, 2002.

S. Li, A. Frisoli, M. Solazzi, and M. Bergamasco. Mechanical design and optimization of a novel fmri compatible haptic manipulator. In *RO-MAN, 2010 IEEE*, pages 1–6. IEEE, 2010.

Y. Li and Q. Xu. Kinematic analysis and design of a new 3-dof translational parallel manipulator. *Journal of Mechanical Design*, 128:729, 2006.

B. Siciliano and O. Khatib. *Springer handbook of robotics*. Springer-Verlag New York Inc, 2008.

L.W. Tsai and S. Joshi. Kinematics and optimization of a spatial 3-upu parallel manipulator. *Journal of Mechanical Design*, 122:439, 2000.

D.R. Walter, M.L. Husty, and M. Pfurner. The snu 3-upu parallel robot from a theoretical viewpoint. In *Proceedings of the Second International Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, pages 1–8, 2008.

L. Wang and J. Xi. *Smart devices and machines for advanced manufacturing*. Springer Verlag, 2008.

Chapter III
Robot Design

Portable Posture Guiding System With Visual and Verbal Feedback for Upper Extremity

Wei-Ting Yang, Ke Yeng Tee, I-Ming Chen,
Albert Causo, and Song Huat Yeo

School of Mechanical and Aerospace Engineering,
Nanyang Technological University, Singapore 639798, Singapore *

Abstract A portable motion replication system for posture guidance of upper extremity is presented in this paper. The system can capture user's postures, compare the postures with corresponding ones in database, and give the user proper visual and verbal feedback for posture modifying in real time. The experiment of user test shows that the subjects under both visual and verbal feedback have better performance.

1 Introduction

In a class of motion learning, a teacher implements three important works: demonstration, observation, and multi-modal feedback. When the student's motion is wrong, the teacher demonstrates key poses, enhances verbal instruction, or adjusts student's motion by touching. However, teaching videos are another choice for people who cannot attend course since sense of sight is a main communication. Though 3-D virtual reality enhances effect of learning (Chua et al., 2003), but users still do not know if their performances are correct. A new concept, called motion replication (MoRep), makes the motion learning from a virtual teacher possible (Ruttkay et al., 2006). A sound MoRep system includes motion capture, motion comparison, and multi-modal feedbacks. The system could send proper feedbacks for reminding the user how to correct movement.

Feedback function is the core of a MoRep system. Vision, audio, and tactus are three main types of communication for motion learning. MoRep

*This work was supported in part by the Agency for Science, Technology and Research, Singapore, under SERC Grant 092 149 0082, and Media Development Authority, Singapore under NRF IDM004-005 Grant.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_16,

systems may adopt various combination of feedback type (Qian et al., 2004; Portillo-Rodriguez et al., 2008). Baek et al. (2003) developed a virtual-reality based training system which contained an avatar motion-guiding interface, motion retargeting, and audio and image-advice feedback. Lieberman and Breazeal (2007) developed a wearable vibrotactile feedback suit for motor training of the right arm. Chan et al. (2011) proposed a dance training system which provided immediate feedbacks to guide students to improve their skills and evaluated the similarity of two motions.

In this paper, a portable MoRep system for posture guidance of upper extremity is developed. An arm suit with two inertial measurement units (IMU) is used for motion capture. There is an intuitive human-computer interface with 3-D simulation environment. The system provides visual and verbal feedback for helping users to achieve required postures faster and to learn efficiently.

2 Visual and Verbal Feedback

2.1 Visual Presentation and Feedback

In our human-computer interface, a corresponding left upper extremity is shown in 3-D virtual reality. The window of interface is split into two parts for different view (see Figure 1(a)). A virtual teacher's posture appears as a fixed goal. The user could achieve the assigned posture by superimposing the teacher's upper extremity. A total score which means the similarity of posture between teacher and user is shown on the top of the window. It ranges between 0 and 100. Larger score means higher similarity. The color of virtual teacher's extremity changes according to similarity of posture (see Figure 1(b)). The color is transparent dark red when two postures are totally different, and it is transparent green when two postures are totally the same. The rest states are presented by gradient colors between red and green. Words for reminding the learner where to move to achieve assigned posture are also shown in the middle of window.

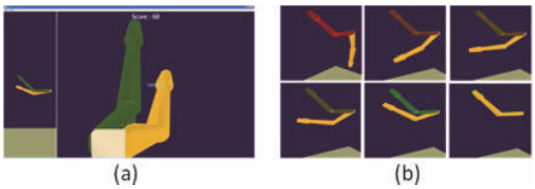


Figure 1. (a) The human-computer interface. (b) The color of teacher's limbs changed by similarity of posture.

2.2 Verbal Instruction

Good verbal instructions are intuitive and correspond with human cognition and learning experience. The left upper extremity could be considered as a five degree of freedom (dof) model which includes shoulder (3 dof), elbow (1 dof) and radius-ulna (1 dof) joint. The adjustment steps and words are defined according to this model. Three steps could make this model achieving any assigned postures: (1) adjust position of wrist, (2) keep wrist position and rotate whole arm, and (3) rotate radius-ulna joint.

Wrist Position. Figure 2(a) shows the diagram of left upper extremities of user and teacher. The 1×3 vector between two wrists is called error vector, V_d. It could be regarded as the direction which the user should move to for achieving assigned posture. The length of error vector means the magnitude of movement. If the length exceeds one-twentieth length of user's upper extremity, the verbal feedback will react. The move of wrist is simplified into six directions: up, down, left, right, forward, and backward. At first, the absolute values of three elements are calculated, and then the maximum element is chosen. According to values of elements of error vector and their sign, proper direction is given as verbal feedback (see Table 1). In order to be intuitive, the instructions are short and clear. The format is the word *hand* plus *direction of adjustment* such as *hand up* and *hand down*.

Table 1. Elements of error vector and corresponding directions.

$\vec{V_d}$=(X_d,Y_d,Z_d)	$+X_d$	$-X_d$	$+Y_d$	$-Y_d$	$+Z_d$	$-Z_d$
Directions	Forward	Backward	Left	Right	Up	Down

Arm Orientation. The step is to fix wrist position and adjust orientation of whole upper extremity (see Figure 2(b)). The positions of two wrists and two shoulders respectively coincide in this step. The dash line is the vector of rotation axis which is from shoulder to wrist. The user could meet required posture by rotating upper extremity about the axis. At first, the similarity score of arm is calculated by comparison method (Yang et al., 2010). If the score is under 90, the verbal feedback will react. The 3-D orientation is noted by Quaternion, q. The required rotation axis and magnitude are obtained by using arm orientations of user, q_{uarm}, and teacher, q_{tarm} (1).

$$\begin{aligned} q_r &= q_{uarm} \cdot (q_{tarm})^{-1} = (w_r, \vec{v}_r) \\ \theta_r &= 2\sin^{-1}(w_r) \end{aligned} \tag{1}$$

Where $\vec{v}_r$ is vector of rotation axis, and θ_r is rotation angle. The verbal instruction is determined by the direction of rotation axis and the sign of rotation angle. Positive $\vec{v}_r$ means its direction is from shoulder to wrist, and negative one means opposite direction. Left part of Table 2 shows the words of instructions in all kinds of conditions.

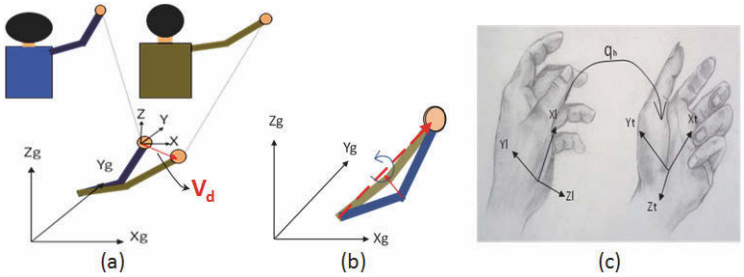


Figure 2. Diagram of two upper extremities at (a) first step, (b) second step, and (c) third step

Table 2. Verbal instructions of arm and hand under different conditions.

Arm orientation			Hand orientation		
$\pm\vec{v}_r$	$\pm\theta_r$	Instructions	$\pm\vec{v}_h$	$\pm\theta_h$	Instructions
+	-	Keep hand in position, rotate elbow inside,	+	-	Rotate hand outside
-	+		+	-	
+	+	Keep hand in position, rotate elbow outside	+	+	Rotate hand inside
-	-		-	-	

Hand Orientation. The third step is to adjust the orientation of radius-ulna joint. In other word, the motion is called pronation and supination of hand (see Figure 2(c)). The total score is calculated by comparison method first (Yang et al., 2010). If the score is under 90, proper verbal instructions are given according to immediate posture. The required rotation axis and magnitude are calculated by using forearm orientations of user, q_{ufarm}, and teacher q_{tfarm}, as following equation:

$$\begin{aligned} q_h &= q_{ufarm} \cdot (q_{tfarm})^{-1} = (w_h, \vec{v}_h) \\ \theta_r &= 2\sin^{-1}(w_h) \end{aligned} \tag{2}$$

Where $\vec{v}_h$ is vector of rotation axis, and θ_h is rotation angle. Positive $\vec{v}_h$ means its direction is from elbow to wrist, and negative one means opposite

direction. Right part of Table 2 shows the words of instructions in all kinds of conditions for left radius-ulna joint.

3 System Implementation

3.1 Hardware and Simulation Environment

The arm suit is composed of two wireless IMU (InertiaCube, InterSense Inc.) and one receiver. The two IMU are attached to the bony end of arm and forearm respectively for reducing move effect on soft tissue (see Figure 3(a)). The IMU1 captures the motion of shoulder joint. The arrangement of IMU2 makes it capable of simultaneously capturing the motion of elbow joint and radius-ulna joint. The human-computer interface is built on Microsoft .NET 3.5 SP1 Framework with Window Presentation Form (WPF). WPF utilizes DirectX for rendering the 3-D simulation environment. Its advanced data binding mechanism enables displaying angles of each joint with processed value from the sensors.

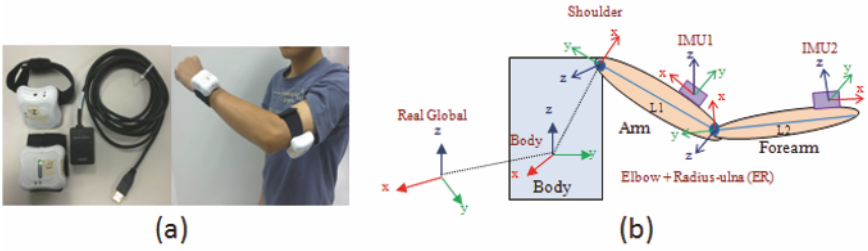


Figure 3. (a) The arm suit and the arrangement. (b) The kinematic model of left upper extremity.

3.2 Kinematic Model

For an IMU-based MoCap system, the calculation of orientation and position are based on a kinematic model. Figure 3(b) shows the simplified kinematic model of left upper extremity with relevant Cartesian coordinate systems (frame). This model includes two linkages (arm and forearm) and three joints (shoulder, elbow, and radius-ulna). The frames of shoulder, elbow and radius-ulna (ER), and body are imaginary in real world, and there are corresponding frames in virtual world. The body frame indicates the facing direction of the user. The real global frame is determined by terrestrial magnetism. In the following calculation, the orientation of frame

is represented by 3×3 rotation matrix, R. The orientations of shoulder and ER frame with respect to real global frame could be calculated according to following equations:

$$\begin{aligned} {}^{RG}R_{Shoulder} &= {}^{RG}R_{Body} \cdot {}^{Body}R_{Shoulder} \\ {}^{RG}R_{ER} &= {}^{RG}R_{Shoulder} \cdot {}^{Shoulder}R_{ER} \end{aligned} \tag{3}$$

The position of shoulder, $P_{Shoulder}$, is defined as an origin, and the positions of elbow and wrist (P_{Elbow} and P_{Wrist}) could be calculated by following equations:

$$\begin{aligned} P_{Shoulder} &= [0 \quad 0 \quad 0] \\ P_{Elbow} &= {}^{RG}R_{Shoulder} \cdot V_1 \\ P_{Wrist} &= {}^{RG}R_{ER} \cdot V_2 + P_{Elbow} \end{aligned} \tag{4}$$

V_1 is a 3×1 vector which means the position of elbow with respect to shoulder frame. V_2 means the position of wrist with respect to ER frame.

4 Experiment and Result

The experiment is designed to evaluate whether the feedbacks improve user's posture learning. Thirty normal and health subjects, whose ages are between 21 and 32 (mean age = 27.03), are selected. They are equally grouped into three groups for three feedback modes. Group1 includes visual and verbal feedback. Group2 only has visual feedback but no reminding words. Group3 does not have feedbacks and superimposition effect. At first, we introduce the subject to the system and human-computer interface. The body of the subject is fixed once finishing the calibration. At the beginning, the subject naturally put upper extremity downwards along body as preparing posture. When an assigned posture shows on the display, the subject starts to imitate it. Once the subject achieves the posture, the subject returns to preparing posture and waits for next assigned posture. The time taken from preparing posture to achieving posture, call passing time, is recorded for assessing the performance of the subjects. There are 4 assigned postures in the experiment, and each posture is repeated 4 times. The assigned postures are shown randomly.

A mean passing time is calculated by using 10 passing time of the same posture at the same time in the same group (Equation 5).

$$T_{jk}^{g} = \frac{\sum_{i} t_{jk}^{i}}{10} \tag{5}$$

Where T is mean passing time, and g is identification of group which ranges between 1 and 3. i means identification of subject which ranges between 1 and 30 and are evenly separated into three groups in order. j is identification of posture which ranges between 1 and 4, and k is number of time which ranges between 1 and 4.

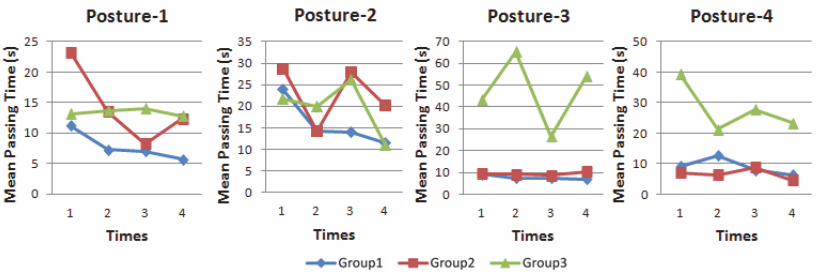


Figure 4. Mean passing time versus number of times under three different groups in four different postures.

Figure 4 shows the charts of mean passing time versus number of times under three different groups in four different postures. The results show that the subjects in Group1 take shorter time than the subjects in other groups. Though the passing time in Group1 is not always the shortest one in each number of times, the sum of the passing time in Group1 is still smaller than those in other groups. It proves that the combination of visual and verbal feedback helps the subjects to learn better. In the charts of Posture1 and Posture2, the mean passing time of Group1 stably decreases when the number of times increases. Thought most subjects feel that Posture-2 is more difficult, the results show that the subjects in Group1 obviously make great progress. However, the performances of the subjects in Group2 and Group3 are unstable and not better than those in Group1. Most subjects feel that Posture-3 and Posture-4 are easier to perform. The results show that the performances of the subjects in Group1 and Group2 do not make any different. It shows that if a posture is easy to perform, visual feedback is enough to help user to learn the posture. In substance, visual and verbal feedbacks obviously improve performance of posture learning.

5 Conclusion

A MoRep system with visual and verbal feedback for posture guidance of upper extremity is implemented. A user could immerse in the 3-D simulation

environment and seamlessly learn new postures from the virtual teacher. In our system, visual feedback is prior for rough posture replication since vision is a main communication for learning movements. The verbal feedback is used for fine posture adjustment, and it reacts only when the user's posture is close to assigned one. The user test proves that the system with both visual and verbal feedback helps the subjects to learn better and faster.

Bibliography

S. Baek, S. Lee, and G. J. Kim. Motion retargeting and evaluation for vr-based training of free motions. *The Visual Computer*, 19:222–242, 2003.

J. Chan, H. Leung, J. Tang, and T. Komura. A virtual reality dance training system using motion capture technology. *IEEE Transaction on Learning Technologies*, 4(2):187–195, 2011.

P. T. Chua, R. Crivella, B. Daly, N. Hu, R. Schaaf, D.Ventura, T. Camill, J. Hodgins, and R. Pausch. Training for physical tasks in virtual environments: Tai chi. In *Proceedings of the IEEE International Conference on Virtual Reality*, pages 87–94, 2003.

J. Lieberman and C. Breazeal. Tikl: Development of a wearable vibrotactile feedback suit for improved human motor learning. *IEEE Transaction on Robotics*, 23(5):919–926, 2007.

O. Portillo-Rodriguez, O. O. Sandoval-Gonzalez, E. Ruffaldi, R. Leonardi, C. A. Avizzano, and M. Bergamasco, editors. *Real-time gesture recognition, evaluation and feed-forward correction of a multimodal Tai-Chi platform - Proceedings of the 3rd International Workshop HAID08,* Jyväskylä, Finland, September 15–16, 2008, Lecture Notes in Computer Science vol. 5270, pp.30–39, 2008. Springer Verlag, Wien-New York.

G. Qian, F. Guo, T. Ingalls, L. Olson, J. James, and T. Rikakis. A gesture-driven multimodal interactive dance system. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1579–1582, 2004.

Z. Ruttkay, J. Zwiers, H. V. Welbergen, and D. Reidsma, editors. *Towards a Reactive Virtual Trainer - Proceedings of the 6th International Conference IVA06,* Marina Del Rey, CA, USA, August 20–22, 2006, Lecture Notes in Computer Science vol. 4133, pp.292–303, 2006. Springer Verlag, Wien-New York.

W. T. Yang, Z. Luo, I. M. Chen, and S. H. Yeo, editors. *A method for comparing human postures from motion capture data - Proceedings of the 19th CISM-IFToMM Symposium ROMANSY10,* Udine, Italy, July 5–8, 2010, CISM Courses and Lectures vol. 524, pp. 441–448, 2010. Springer Verlag, Wien-New York.

An Innovative Actuation Concept for a New Hybrid Robotic System

Basilio Lenzo, Antonio Frisoli, Fabio Salsedo and Massimo Bergamasco*
PercRo Lab - TeCIP Institute - Scuola Superiore Sant'Anna, Pisa, Italy

Abstract In this article a novel actuation concept of robotic manipulators is presented. The concept is based on an innovative extension of gravity balancing techniques using passive elements. It allows to exert, at the level of the end effector, a force of variable intensity and generic spatial orientation without any motor torque required during any movement of the manipulator, except when a change of the exerted force's intensity and/or orientation is requested. So, a new hybrid machine can be realized combining the benefits of active and passive existing systems without their drawbacks, with possible very high potentials in robotics applications.

1 Introduction

Nowadays a great number of applications involve robotic manipulators with different kinematics which have to generate constant forces for relatively long times (Lan et al., 2010 and Lan and Wang, 2011). There are basically two approaches to this kind of problem: active and passive robotic systems. Active systems are spreadly diffused, their major advantage is the totally free choice of kinematics, however since each joint is actuated there are significant energy consumptions and the need of sophisticated controls, with use of a large number of sensors (Duysinx and Geradin, 2004 and Kikuchi et al., 2009). This architecture is not intrinsically safe since, as an example, the failure of a sensor could lead to instability. Passive systems instead do not entail any energy consumption, but their kinematics cannot be arbitrary and, being passive, they need a manual tuning (Collins et al. (2001, 2005)).

This article presents a novel actuation concept for serial robot manipulators, theoretically based on an innovative extension of gravity compensation techniques, with the aim of creating a new hybrid machine characterized

*This work has been supported by the European Project VERE, an Integrated Project funded under the European Seventh Framework Program, Future and Emerging Technologies (FET), Grant Agreement Number 257695.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_17,

by the benefits of both active and passive systems, without including their shortcomings. First, the state of the art of gravity balancing of serial manipulators is analyzed; then, a new theoretical approach for gravity balancing is proposed. Subsequently, the new actuation concept is developed.

2 Gravity Balancing: Basics and Extensions

A machine is said to be gravity balanced if no joint actuator inputs are needed to keep the system in equilibrium for any configuration, being the potential energy of the device invariant. Over the years, several methods of gravity balancing have been proposed (Fattah and Agrawal, 2006), exploiting clever designs using counterweights (Hayward et al., 1998 and Massie and Salisbury, 1994), auxiliary parallelograms and springs (Agrawal and Fattah, 2004, Barents et al., 2009, Lin et al., 2010).

Suppose to have a weightless link which can rotate towards point O and has an attached mass m at the other extremity (Figure 1a). This system can be gravity balanced using a zero free-length spring whose extremities are connected respectively to the link at point Q, distant b from point O, and to another point P which is distant a from point O in the vertical direction, opposite to gravity. Its stiffness k must be (Barents et al., 2009):

$$k = \frac{mgl}{ab} \tag{1}$$

In the literature the majority of spring balanced mechanisms are planar. There have been efforts to extend this approach to spatial mechanisms: Agrawal and Fattah (2004) present a hybrid method but it is complicated and too bulky to be implemented on an orthosis. The approach we present here is inspired to a more recent work (Lin et al., 2010): we propose a simpler mathematical approach, and then an extension of the method.

Consider a generic n-DOF serial manipulator and its generic link i (Figure 1b). Let us assume the following notation:

- $\vec{g}$ the gravity acceleration vector;
- $\vec{r}_0$ a vertical unit vector, opposite to $\vec{g}$;
- O_{i-1} the point of connection between link $i-1$ and link i;
- O_i the point of connection between link i and link $i+1$
- $\vec{x}_i$ a unit vector defining the spatial direction of link i, going from point O_{i-1} to point O_i, according to the Denavit-Hartenberg convention;
- m_i the mass of link i, concentrated in point P_i distant h_i from O_{i-1};
- l_i the overall length of link i;
- $\vec{P}_i$ the vector connecting point O_0 with point P_i.

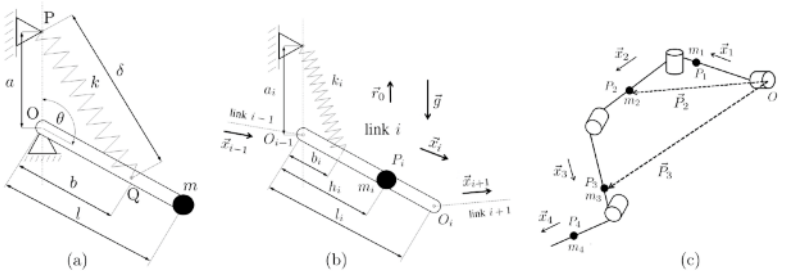


Figure 1. Single link configuration (a); generic link i, values and vector of interest (b); part of a generic spatial manipulator (c).

A spring of stiffness k_i is attached to link i, being the third side of a triangle of which one side is $\vec{b}_i = b_i\vec{x}_i$ and the other is $\vec{a}_i = a_i\vec{r}_0$. It is important to notice that b_i is directed along $\vec{x}_i$ so it depends on how the link is oriented in space, depending on the configuration of the manipulator, while a_i is supposed to be *always* directed as $\vec{r}_0$.

We now show that it is possible to make the overall energy of the system constant independently of the configuration. The potential energy of the masses is $E_m = \sum_{i=1}^{n} m_i g\vec{P}_i \cdot \vec{r}_0$, the elastic energy of the springs is $E_s = \sum_{i=1}^{n} \frac{1}{2}k_i(||\vec{b}_i - \vec{a}_i||)^2 = \sum_{i=1}^{n}\left(\frac{1}{2}k_i(a_i^2 + b_i^2) - k_i\vec{a}_i \cdot \vec{b}_i\right)$ but since $\vec{b}_i = b_i\vec{x}_i$, $\vec{a}_i = a_i\vec{r}_0$ we have $E_s = \sum_{i=1}^{n}\left(\frac{1}{2}k_i(a_i^2 + b_i^2) - k_i a_i b_i \vec{x}_i \cdot \vec{r}_0\right)$.

The vector $\vec{P}_i$ can be written as a sum of vectors, beginning from point O and proceeding link by link along each $\vec{x}_i$: $\vec{P}_i = \sum_{j=1}^{i-1} l_j\vec{x}_j + h_i\vec{x}_i$ for $i > 1$ (see Figure 1c). Thus, we get

$$E' = E_m + E_s - \frac{1}{2}k_i(a_i^2 + b_i^2) = \sum_{i=1}^{n}\left(m_i g \sum_{j=1}^{i-1} l_j\vec{x}_j \cdot \vec{r}_0 + m_i g h_i \vec{x}_i \cdot \vec{r}_0 - k_i a_i b_i \vec{x}_i \cdot \vec{r}_0\right) \tag{2}$$

where we subtracted the term $\frac{1}{2}k_i(a_i^2 + b_i^2)$, already constant. Now it is possible to group all terms containing the inner product $\vec{x}_i \cdot \vec{r}_0$ for each i, and *impose* their sum to be zero: we get $E' = 0$ and therefore $E_m + E_s$ will be *a constant independently of the configuration of the manipulator*, since nothing has been said about it. Finally, springs have to be designed as

$$k_i = \frac{m_i g h_i + \sum_{j=i+1}^{n} m_j g l_j}{a_i b_i} \tag{3}$$

3 Innovative Actuation Concept: An Example

So far, we have built a system that balances gravity forces, applicable to manipulators of whatever kinematics. From a more general point of view, we are able to balance *any* force whose direction is the same of gravity, but this is only related to the choice of the vector $\vec{r}_0$. If we modify this direction with any other, let us denote it with the unit vector $\vec{p}$, we would be able to balance forces directed as $\vec{p}$, and more in general we would be able to *exert forces directed as* $\vec{p}$. More formally, what we just claimed would be easy to show with a little rearrangement of the demonstration carried out before.

Based on the above, we can design only *one* system of springs, for which loads and their direction[1] are known a priori: consequently, the force exerted has fixed direction and fixed intensity. The innovative concept we present here overcomes these two limits allowing to exert forces of different intensity and direction, making possible to think to an *actuation system*.

For sake of clarity, we propose an example which we will develop on the way. Suppose to have a serial RR manipulator whose links are weightless and for which it is requested to apply on its end effector a vertical force opposite to gravity (Figure 2a). In order not to affect the workspace of the manipulator, we put the actuation system *remotely*: it is therefore necessary to *replicate* the orientation of the links of the manipulator (needed from the extended balancing theory). So, we connect each joint of the robotic manipulator via tendon transmission with a mechanism called *mirror mechanism*, which replicates the exact orientation of each link.

As explained previously, the spring k_i relative to link i is connected to a segment oriented as the link i, and to another one oriented as the desired direction of the force to exert: obviously this direction must not vary with any movement of the links. Therefore, it has been chosen to make the replicated links not as a serial chain as in the limb, but independent each of other, all connected to a *fixed point* called *center of mechanism*. This gives indeed the possibility of having, for each spring, a fixed extremity regardless of the configuration of any link of the manipulator, as requested in the demonstration previously carried out. This extremity is fixed and located onto a segment long a_i and oriented in the space as $\vec{p}$ in general, while it will be vertical in our example.

However, in the case in analysis the two axes of the robot are incident in one point, therefore the position of the end effector respect to the ground (involved in the potential energy) is defined only by the segment L_1 (Figure 2), then we need only *one* spring connected to the replicated link corresponding to the last link of the RR robot.

[1] Which clearly has to be the same for all loads.

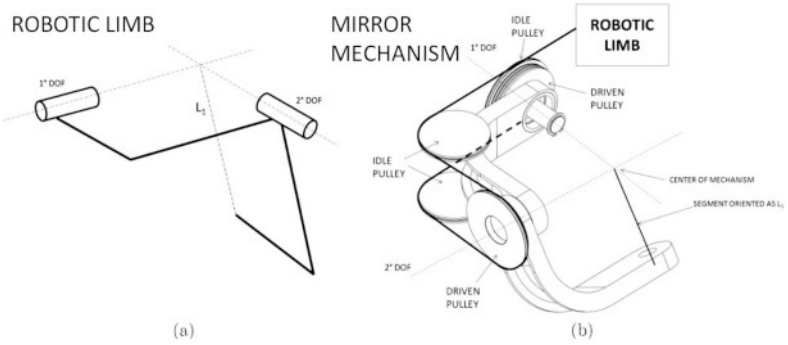


Figure 2. Example of kinematics (a) and relative mirror mechanism (b).

The design of the system is done using Eq. 3 which reduces to $k_i = \frac{mgl}{a_i b_i}$ since we considered weightless links and an only mass m attached to the end effector; in the general case of an applied force F, we have $k_i = \frac{Fl}{a_i b_i}$. In the example considered, $k = \frac{Fl}{ab}$. Once the system has been designed, k and b are fixed, and if we could change the value of a in some way with a motor, we could adjust it in order to exert different forces:[2] $a' = \frac{F'l}{bk}$. The *passive torque generator* (Figure 3a) performs the implementation of the zero free-length spring with a known system consisting of cable, a real spring and three pulleys; a motor drives one of the pulleys which can translate in order to adjust the value of a. Now, suppose that the spatial orientation of the required force changes: if we had a way to modify the orientation of the fixed segment (of length a) of the mirror mechanism, defined from the unit vector $\vec{p}$, we would be able to exert forces in any direction. A generic orientation is obtained using two motors, each working on a different axis, implemented on a *orientation mechanism* (Figure 3b).

All the presented ideas are suitably merged in Figure 4. To sum up, the architecture developed for a generic robot manipulator (for instance, a limb of an exoskeleton) provides the following main components:

- the robotic manipulator itself, whose joints are mechanically connected to the remote actuation system via tendon transmission;
- a *mirror mechanism* which replicates the orientation of each link of the robotic limb;
- a number of *passive torque generators* having a *spring* (one each) and a motor (one for all) for changing the intensity of the force to exert;

[2] In a general case with more than one spring, in order to use only one motor, we choose to set a_i to be the same for each i.

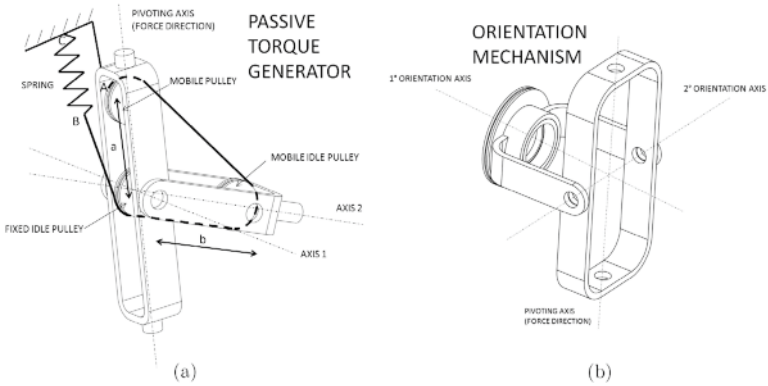


Figure 3. Passive torque generator (a) and orientation mechanism (b).

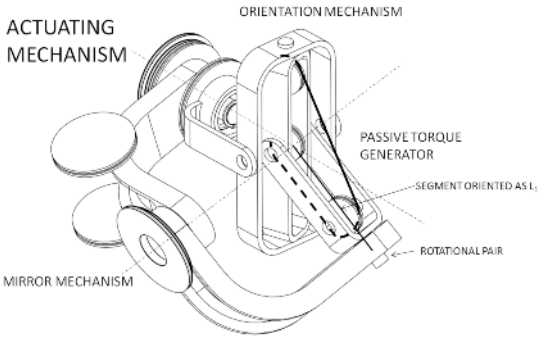


Figure 4. Actuation mechanism developed.

- an *orientation mechanism* driven by two motors which allows to modify the orientation of the force to exert at the end effector.

The chosen architecture implies that the torques needed by the robotic limb to exert the required force during the user movement are provided by the springs and not by motors. So, the motors are used *only* when it is required to change the intensity and/or orientation of the applied force, hence with a great save of energy. Another benefit is that the chosen configuration grants a minimum inertia of the robotic limb and does not restrict its workspace, since the limb consists only of its links and relative joints and the actuation system is situated remotely. Moreover, being motors controlled in position and not in force, stability problems are intrinsically avoided.

Now we show an example of operation of the system analyzed. We assume the lagrangian coordinates θ_1 and θ_2 to be zero in the configuration shown in Figure 5a. The Denavit-Hartenberg parameters are

Link	a_i	α_i	d_i	θ_i
1	0	$-\pi/2$	0	$\pi + \theta_1$
2	L_1	0	0	θ_2

Once assigned a motion law to θ_1 and θ_2, we can study the trend of potential energy, elastic energy and force exerted by the spring during the motion of the manipulator. Figure 5b shows an example in which θ_1 and θ_2 change from 0 to respectively $\frac{5\pi}{4}$ and $\frac{\pi}{2}$ in 30 seconds: it is clear how potential energy and elastic energy vary mantaining their sum constant.

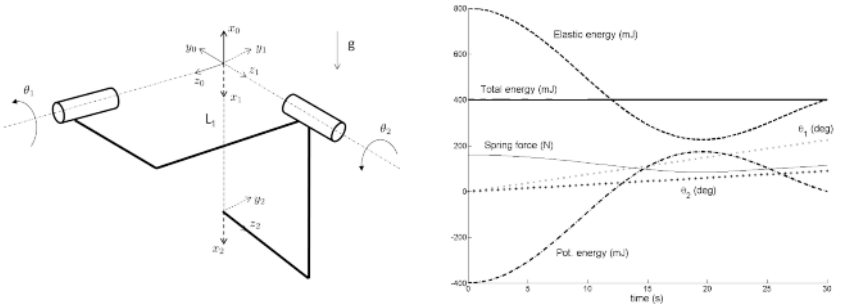


Figure 5. Kinematics and coordinate systems (a); results of the simulation (parameters: $a = b = 5$ mm, $F = 40$ N, $L_1 = 10$ mm) (b).

4 Conclusion

We proposed an innovative actuation concept (which is patent pending) for the design of a new hybrid robotic system. It is based on a novel extension of known gravity compensation techniques exploiting passive elements (springs). The new concept implies that no motor torque is needed during any user movement, except when a change of the applied force's intensity and/or orientation is required. A novel light-weight exoskeleton, exploting our new concept on an anthropomorphic kinematics, is now under development. Details about its design have not been mentioned here for space reasons, and they will be presented in future articles.

Bibliography

S.K. Agrawal and A. Fattah. Gravity-balancing of spatial robotic manipulators. *Mechanism and machine theory*, 39(12):1331–1344, 2004.

R. Barents, M. Schenk, W.D. van Dorsser, B.M. Wisse, and J.L. Herder. Spring-to-spring balancing as energy-free adjustment method in gravity equilibrators. ASME, 2009.

S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082, 2005.

S.H. Collins, M. Wisse, and A. Ruina. A three-dimensional passive-dynamic walking robot with two legs and knees. *The International Journal of Robotics Research*, 20(7):607–615, 2001.

P. Duysinx and M. Geradin. An introduction to robotics: Mechanical aspects. *University of Liège*, 2004.

A. Fattah and SK Agrawal. Gravity-balancing of classes of industrial robots. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2872–2877. IEEE, 2006.

V. Hayward, P. Gregorio, O. Astley, S. Greenish, M. Doyon, L. Lessard, J. McDougall, I. Sinclair, S. Boelen, X. Chen, et al. Freedom-7: A high fidelity seven axis haptic device with application to surgical training. *Experimental Robotics V*, pages 443–456, 1998.

T. Kikuchi, Y. Jin, K. Fukushima, H. Akai, and J. Furusho. Study on development of active-passive rehabilitation system for upper limbs: Hybridplemo. In *Journal of Physics: Conference Series*, volume 149, page 012016. IOP Publishing, 2009.

C.C. Lan and J.Y. Wang. Design of adjustable constant-force forceps for robot-assisted surgical manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 386–391. IEEE, 2011.

C.C. Lan, J.H. Wang, and Y.H. Chen. A compliant constant-force mechanism for adaptive robot end-effector operations. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2131–2136. IEEE, 2010.

P.Y. Lin, W.B. Shieh, and D.Z. Chen. Design of a gravity-balanced general spatial serial-type manipulator. *Journal of Mechanisms and Robotics (Transactions of the ASME)*, 2(3), 2010.

T.H. Massie and J.K. Salisbury. The phantom haptic interface: A device for probing virtual objects. In *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*, volume 55, pages 295–300, 1994.

Development of 3-Axis Flexure Stage for Bio-Cellular Experimental Apparatus

Daisuke Matsuura* and Chia-Hsiang Menq†

* Department of Mechanical Sciences and Engineering, Tokyo Institute of Technology, Japan

† Department of Mechanical and Aerospace and Engineering, The Ohio State University, USA

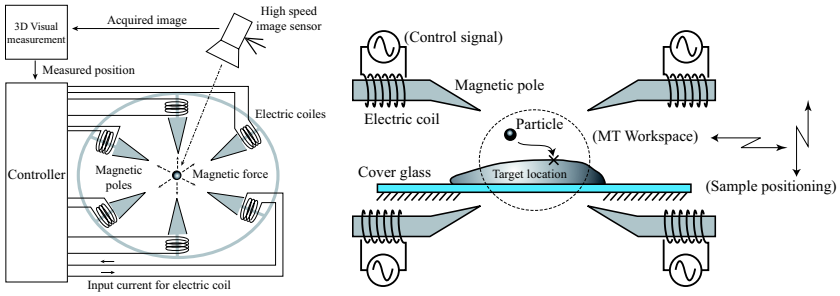
Abstract The authors discuss the design and implementation of a precision flexure mechanism for a non-contact bio-cellular experimental apparatus called magnetic tweezer (MT). MT manipulates a magnetic particle of several-micron diameter by magnetomotive force, diverged from magnetic poles to stimulate a target cell to measure its mechanical property. The developed flexure stage was designed to satisfy a strict size limitation to be assembled in a conventional microscope, and to make MT a practical experimental apparatus. The design concept and series of analysis were discussed to satisfy practical requirements on working range and mechanical stiffness. Motion control experiment using a 3D-visual servoing was performed to evaluate the control accuracy and to demonstrate validity and effectiveness of the developed system.

1 Introduction

In order to evaluate mechanical property of live cells to investigate their living mechanism, high precision microscopic manipulation system which can give external force and can observe resulting deformation to calculate mechanical properties is strongly needed. In biophysical research field, many research activities thus have been devoted. Since Crick et al.(1) utilized microscopic magnetic particles to characterize the physical properties of the cytoplasm of chick fibroblasts, magnetic tweezer (MT), which is illustrated in **Fig.1(a)**, has been studied (2)-(5), and mathematical model was also formulated to establish forward and inverse force control (6),(7). However, from the point of view to use MT for practical bio experiment, some problems are still exist. Firstly, typical working range of MT is very small, several hundred micro meters, although specimen cells are randomly distributed on a culture dish of several tenth millimeters. A precise sample motion stage is

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_18,

therefore necessary for specimen cell positioning as illustrated in **Fig.1(b)**. Another problem is mechanical drift. To hold a sample cell at center of MT workspace during long time experiment, the sample motion stage will also be useful. This paper will propose design and implementation of above mentioned mechanism in actual 3D-MT apparatus. Small size and high stiffness will be significantly required, since available volume to put the mechanism is strictly limited in order to avoid conflict between the stage and existing optical microscope and 3D-MT components. Motion control of the stage will be established based on visual servoing technique using a high-speed CMOS image sensor, to demonstrate the validity and effectiveness of the developed system.



(a) Schematic of 3D-MT (b) Cooperative operation of 3D-MT and sample motion stage

Figure 1. Schematic of 3D-MT and cooperation with sample motion stage

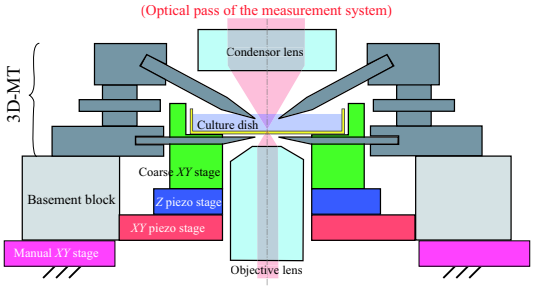


Figure 2. Component of the Mechanism

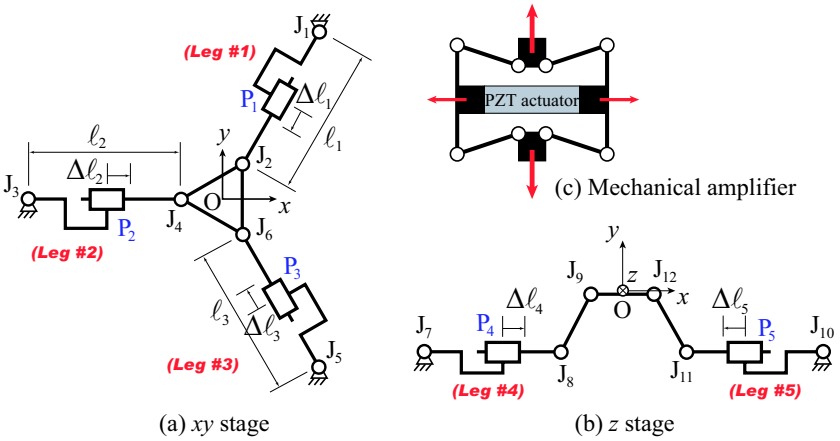


Figure 3. Schematic of the Mechanism

2 Design of the Stage Mechanism

2.1 Synthesis of the Mechanism

The motion stage should achieve four tasks; (1) Alignment of the 3D-MT working space against visual measurement system (8), (2) Selecting a target cell among spread ones on a culture dish, (3) Calibration of out-of-plane visual measurement, and (4) Compensation of drift during a long time experiment. To satisfy these four requirements, the sample stage is composed of four different stages illustrated in **Fig.2**; manual compliant stage, piezo-driven compliant *xy* stage and *z* stage, and coarse *xy* motion stage driven by DC servo motors. These stages are designed to be assembled on an inverted microscope Olympus IX81, the optical system for the visual measurement. To assemble the setup, the most critical dimensional constraint is the gap distance between microscope's objective lens and condenser lens, which is 45 mm. To equip the 3D-MT, motion stage and sample holder should be placed within this space. The framework of the stage mechanism was figured out as shown in **Fig.2**.

2.2 Inverse Kinematics Analysis

The compliant mechanism has precision *xy* stage and z stage in the second and third level in **Fig.2**. As illustrated in **Fig.3(a)**, moving plate of the *xy* stage is supported by three legs of same length having prismatic actuator units, which is actuary a piezo electric actuator with a mechanical

amplifier using lever mechanism shown in **Fig.3(c)**, of which amplification ratio is 5:1. Regarding this feature, inverse kinematics can be easily derived. Firstly, displacement of i-th (i=1,2,3) actuator, P_i, can be assumed to have linear relationship against input voltage as

$$\Delta\ell = kV_i \tag{1}$$

where k is a coefficient with respect to the mechanical amplifier. Relationship between desired displacement of the moving plate, $(\Delta x_d, \Delta y_d)$, and actuators' displacement, $(\Delta\ell_{1,2,3})$, can be written as

$$\left\{\begin{array}{lcl} \Delta\ell_1 &=& -\cos\frac{\pi}{3}\Delta X_d - \sin\frac{\pi}{3}\Delta y_d \\ \Delta\ell_2 &=& \Delta X_d \\ \Delta\ell_3 &=& -\cos\frac{\pi}{3}\Delta X_d + \sin\frac{\pi}{3}\Delta y_d \end{array}\right. \Rightarrow$$

$$\begin{bmatrix} \Delta\ell_1 \\ \Delta\ell_2 \\ \Delta\ell_3 \end{bmatrix} = \begin{bmatrix} -\cos\frac{\pi}{3} & -\sin\frac{\pi}{3} \\ 1 & 0 \\ -\cos\frac{\pi}{3} & \sin\frac{\pi}{3} \end{bmatrix} \begin{bmatrix} \Delta x_d \\ \Delta y_d \end{bmatrix} \tag{2}$$

By substituting eq.(1) into eq.(2), command voltage for actuators is determined as

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} V_{1,0} + \Delta V_1 \\ V_{2,0} + \Delta V_2 \\ V_{3,0} + \Delta V_3 \end{bmatrix} + \frac{1}{k} \begin{bmatrix} -\cos\frac{\pi}{3} & -\sin\frac{\pi}{3} \\ 1 & 0 \\ -\cos\frac{\pi}{3} & \sin\frac{\pi}{3} \end{bmatrix} \begin{matrix} c \\ \end{matrix} \begin{bmatrix} \Delta x_d \\ \Delta y_d \end{bmatrix} \tag{3}$$

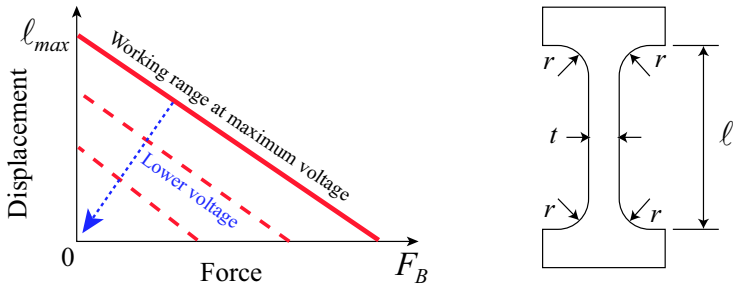
Where $V_{i,0}$ is initial voltage. Since typical piezo actuators do not allow negative input voltage, it is necessary to give initial voltage around the middle of the working range. **Fig.3(b)** illustrates the schematic of the z stage. This is an antagonistic toggle mechanism which converts opposing input displacements of the two prismatic actuators to perpendicular output displacement of the moving plate. Actuators P_1 and P_2 are thus driven synchronously to balance their input force. Since the angle of the link J_2-J_3 and J_5-J_6 is shallow, the mechanism is near-singular condition and therefore the in-plane to out-of-plane displacement conversion can be done in high-efficiency. Due to above mentioned design concept, input voltage and output displacement can be assumed to have linear relationship. For two actuators, same input voltage can be given to achieve a desired out-of-plane displacement as

$$V_{i=4,5} = k\Delta z_d + V_{i,0} \tag{4}$$

where $\mathrm{V}_{i,0}$ is initial voltage, typically defined as the half of maximum input voltage, same as eq.(3).

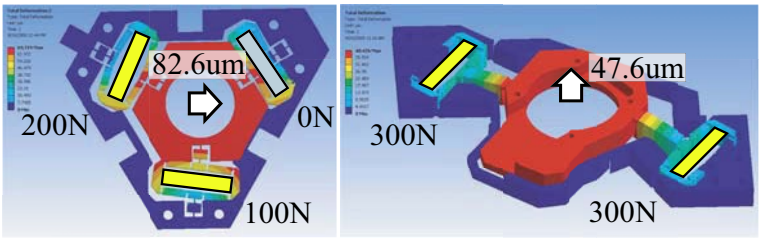
3 Implementation

Design of the 3-axis stage has been decided through mechanical analysis using finite element method (FEM). **Fig.4** shows some examples of FEM results through displacement analysis, giving input force of piezo actuators to investigate resulting output displacement. In this simulation, there is a well-known constraint that output stroke and output force of piezo actuator should be within a triangular area surrounded by origin, maximum stroke ℓ_{max}, and blocking force F_B, as illustrated in **Fig.4(a)**. During the simulation, applied force and displacement of the input link were thus investigated and balanced to satisfy this constraint. Based on this simulation, three design parameters of flexure hinges illustrated in **Fig.4(b)**; thickness t, neck-length ℓ and corner radius r were optimized so that desired working range can be achieved while maintaining enough joint stiffness. As the result, these parameters were determined as r=0.5mm, t=0.5mm and ℓ=1.7mm. By cooperatively operating multiple actuator units according to inverse kinematics solution in eqs.(3) and (4), desired traveling range, 200μm and 100μm for xy and z stage respectively, can be achieved. **Fig.5** shows displacement on x-axis and z-axis. Static load analysis and modal analysis have also been performed to avoid large fluctuation due to external load and resonance, especially at low frequency. **Tab.1** lists the obtained resonant frequencies from the modal analysis. It can be seen that the first mode lower than 200Hz frequency is rotational, not translational, so they will not cause significant vibration due to the control frequency of 200Hz. It can therefore be said that the designed mechanism can achieve highly precise manipulation in the order of nano meter. Designed components were fabricated by wire-cut electrical discharge machining (WEDM). **Fig.6(a)**



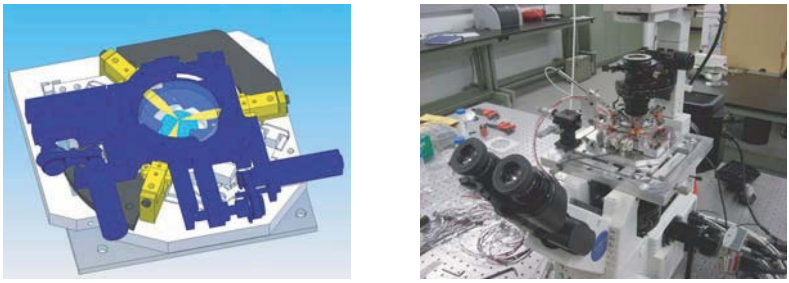
(a) Force-displacement relationship of PZT actuator (b) Optimized design parameters of flexure hinge

Figure 4. Considered factors regarding PZT actuation and flexure hinge



(a) x-axis displacement (b) z-axis displacement

Figure 5. Displacement analysis via FEM



(a) The coarse stage mounted on the PZT stage (b) Assembled setup on IX81 optical microscope

Figure 6. Assembled setup on an inverted microscope

shows the coarse xy stage, which is driven by two DC motors, and lower three magnetic poles, assembled on the developed PZT stage. The coarse stage achieves 8mm×8mm working range, and can be used to select a specimen cell. **Fig.6(b)** shows the assembled entire setup.

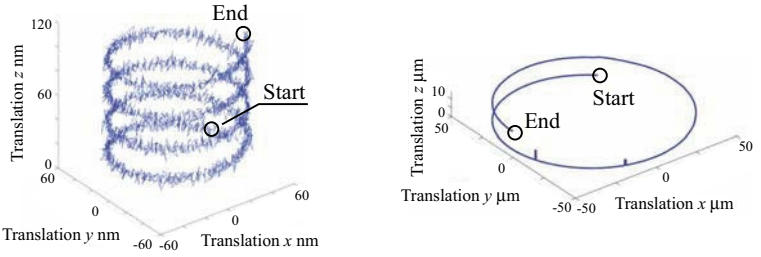
4 Experimental Validation

The developed apparatus was operated by a computer control system, composed of PC-based workstation, having 8-core processor and 4GB RAM, and including DACs (NI PCI-6733 and Measurement Computing PCI-DAC6703), ADC (Measurement Computing PCI-DAS6032) and image grabbing/processing card (Matrix Odyssey XCL), inverted optical microscope Olympus IX81, high-speed CMOS image sensor Mikrotron MC1310 and two power amplifiers to drive the MT and the piezo stage. Developed control software runs each apparatus (MT, piezo stage and microscope) via communications through analog signal on ADC and DAC, and serial protocols, RS-232 and

Table 1. Obtained modal frequency by FEM modal analysis

	1st [Hz]	2nd [Hz]	3rd [Hz]
Manual xy stage	121 (R)	185 (T)	228 (R)
Piezo xy stage	154 (R)	267 (T)	355 (R)
Piezo z stage	241 (T)	293 (T)	392 (R)

* (R): Rotational mode, (T): Translational mode



(a) Spiral trajectory of 100nm diameter and 25nm pitch (b) Spiral trajectory of 100 µm diameter and 10 µm pitch

Figure 7. Measured output trajectories

CameraLink. Using this control system, motion control experiment of the developed 3-axis stage was performed. A target particle of 4.5 μm was attached on the moving plate so that it can be tracked by 3-axial visual measurement system (8), and closed-loop position control can be established. **Fig.15** plots measured output trajectories corresponding to two different references. **Fig.15(a)** is a large trajectory of 100 μm diameter and 10 μm pitch, and **Fig.15(b)** is a sub-micrometer trajectory of 100 nm diameter and 25 nm pitch. In both case, desired trajectories were well achieved, since its fluctuation was within $\pm$10 nm in x and y axis, and within $\pm$15 nm in z axis. Standard deviation of tracking errors on x-y and z aixs were 2.31 nm and 2.00 nm, respectively.

5 Conclusions

A mechatronics system having 3D-MT and 3-axis piezo stage for bio experiment was designed and fabricated. Based on proposed design concept and FEM analysis, an effective stage design was figured out to satisfy a number of requirements aiming for practical use could be satisfied. Output stroke

of the stage covers 200 μm diameter in x-y axis, and 150 μm in z axis. First order mode frequency was higher than 100 Hz, and it did not cause significant resonance. By experimental validation, position control accuracy of the developed stage was validated Using closed-loop motion control with 3D visual measurement. Static positioning error of the piezo stage was $\pm$10 nm in x and y axis, and within $\pm$ 15 nm in z axis, and trajectory tracking of 100 nm diameter and 25 nm pitch was achieved.

Acknowledgment

The work presented in this paper was done by D. Matsuura when he worked as a postdoctoral researcher in the department of Mechanical and Aerospace Engineering at The Ohio State University. It was supported in part by National Institutes of Health under Grant No. R21RR024435.

Bibliography

[1] E. H. C. Crick and A. F. W. Hughes, "The physical properties of cytoplasm: a study by means of the magnetic particle method", Exp. Cell Res., 1, pp. 36-80, 1949.

[2] C. Haber and D. Wirtz, "Magnetic tweezers for DNA micromanipulation", Rev. Sci. Instrum., 71(12), pp. 4561- 4570, 2000.

[3] B.G. Hosu, K. Jakab, P. Banki and F. I. Toth, "Magnetic tweezers for intracellular applications," Rev. Sci. Instum., 74(9), pp. 4158-4163, 2003.

[4] F. J. Alenghat, B. Fabry, K. Y. Tsai, W. H. Goldmann and D. E. Ingber, "Analysis of cell mechanics in single vinculin-deficient cells using a magnetic tweezer," Biochem. Biophys. Res. Commun., 277, pp. 93-99, 2000.

[5] A. H. B. de Vries, B. E. Krenn, R. van Driel, and J. S. Kanger, "Micro magnetic tweezers for nanomanipulation inside live cells," Biophys. J., vol. 88, no. 3, pp. 2137-2144, 2005.

[6] Z. Zhang, K. Huang and C.-H. Menq, "Design, Implementation, and Force Modeling of Quadrupole Magnetic Tweezers," IEEE/ASME Transactions on Mechatronics, vol.15, no.5, (2010), pp.704-713.

[7] Z. Zhang and C.-H. Menq, "Design and Modeling of a 3-D Magnetic Actuator for Magnetic Microbead Manipulation," IEEE/ASME Trans. on Mechatronics, Vol. 16, No. 3, pp.421-430, 2011.

[8] Z. Zhang and C.-H. Menq, "Three-dimensional particle tracking with subnanometer resolution using off-focus images," Appl. Opt., vol. 47, no. 14, pp. 2361-2370, 2008.

Development of Expressive Robotic Head for Bipedal Humanoid Robot with Wide Moveable Range of Facial Parts and Facial Color *

Tatsuhiro Kishi† , Takuya Otani† , Nobutsuna Endo† , Przemyslaw Kryczka† ,
Kenji Hashimoto† , Kei Nakata† , Atsuo Takanishi †‡

† Department of Modern Mechanical Engineering, Waseda Univerity, Japan
‡ Humanoid Robotics Institute (HRI), Waseda University, Japan

Abstract This paper describes the development of a new expressive robotic head for bipedal humanoid robot. Through a preliminary experiment, the authors defined the representative facial expressions for 6 basic emotions. To realize these facial expressions, 24-DoFs of mechanisms that allow wide moveable range, and facial color were needed on the face. We designed compact mechanisms that fit into the head of which major dimensions are based on average of adult Japanese female's size. We conducted a questionnaire surveys to evaluate the facial expression recognition rates. The results show that facial expression recognition rates for 6 basic emotions were increased compared to the old head.

1 Introduction

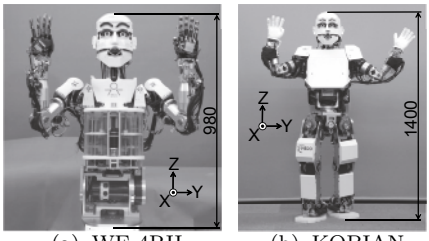
In the future, robots supporting human life should be highly needed. We think that communication between human and robot is important for supporting human life. When humans communicate with robots, the understanding of robots' inner state make the communication more natural.

*This study was conducted as part of the Research Institute for Science and Engineering, Waseda University, and as part of the humanoid project at the Humanoid Robotics Institute, Waseda University. It was also supported in part by RoboSoM project from the European FP7 program (Grant agreement No.: 248366), Global COE Program "Global Robot Academia" from the Ministry of Education, Culture, Sports, Science and Technology of Japan, SolidWorks Japan K.K., NiKKi Fron Co., Chukoh Chemical Industries, DYDEN Corporation and STMicroelectronics, whom we thank for their financial and technical support.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_19,

Therefore, robots should have emotional expression ability. In addition, robots supporting human life should be able to move in the human environment like climbing the stairs or walking on the uneven surface. Bipedal walking is very effective in human environment. Some robots have high facial expression ability supported by the evaluation while others can realize the bipedal walking. For instance, facial expression ability of SAYA (T. Hashimoto et al. (2006)) was evaluated and its facial recognition rates are high. Geminoid (H. Ishiguro and S. Nishio (2007)) has human-like facial expression ability but they are not mobile robots. ASIMO (Y. Sakagami et al. (2002)) is a bipedal walking robot, however it does not have facial expression ability. HRP-4C (K. Kaneko et al. (2009)) has facial expression ability and is a bipedal walking robot, but the facial expression ability has not been evaluated. No currently existing robots have both facial expression supported by assessment and bipedal walking capability. Since 1998 we also have been developing robots which have emotional expression ability. Miwa et. al developed WE-4 series (H. Miwa et al. (2002)) (Fig. 1(a)) which can only express emotions with the upper body. Since 2007, we have been developing emotional expression bipedal humanoid KOBIAN (N. Endo et al. (2008)) (Fig. 1(b)) based on WE-4RII (K. Itoh et al. (2006)) and WABIAN-2R (Yu Ogura et al. (2006)). While designing the robot, we faced some issues. Contrary to WE-4 which had large control devices out of its body, KOBIAN however is a stand-alone robot. Therefore control units like motor drivers should be placed in the robotic head. As the previous version of motor driver was designed for high current motors, its large size did not allow us to design KOBIAN's head as small as the human head. The current size of KOBIAN's head is larger than a human head. Moreover DoFs of KOBIAN's head were decreased from 24 to 7.

Our study found low facial recognition rates for some emotions because the facial expression capability of KOBIAN's head is not sufficient (N. Endo et al. (2008)). KOBIAN expressed whole body emotional expressions, but social robots in environment cannot always express their emotion with their whole body. For example when they carry something in their hands or they handle tasks. Therefore facial expression ability of KOBIAN's head should



(a) WE-4RII (b) KOBIAN

Figure 1. Emotional expression humanoid robots.

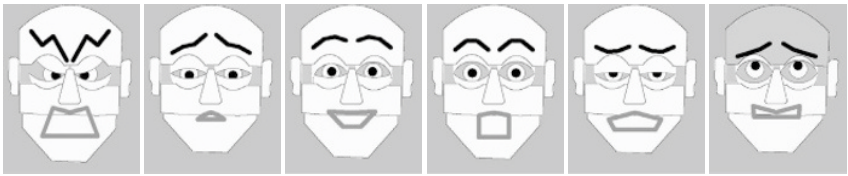
be increased.

In order to realize a robot capable of facial expression easy to understand and bipedal walking, we decided to develop a new robotic head that has the following functions. First, the new head should be able to express the 6 basic emotions (anger, sadness, happiness, surprise, disgust, and fear) supported by Ekman (P. Ekman and W.V. Friesen (1987)). In addition, the size of the new robotic head should adapt to the body of KOBIAN which is designed based on the average size of the adult Japanese female.

Through a preliminary experiment, we obtain representative facial expressions for 6 basic emotions. To realize these facial expressions on robotic head, compact mechanisms for 24-DoFs were developed. Finally, we integrated the head KOBIAN's body. We named the robot KOBIAN-R (KOBIAN-Refined) and its facial expression recognition rates increased 25% in average in the experimental evaluation.

2 Mechanisms

For the development of the new head, we first conducted a preliminary experiment to search the most representative facial expressions on KOBIAN's case. We asked amateur cartoonists to draw 6 basic emotions in Computer Graphics (CG). We conduct a questionnaire with those CGs. In that experiment, we found facial expressions for each emotion that have the highest facial recognition rates (Fig. 2, Fig 3). We found that we need to implement 24-DoFs on KOBIAN-R's head to realize all the synmetrical and asymmetrical facial expressions. These DoFs are shown in Fig. 4 and Table 1. In



(a) Anger (b) Sadness (c) Happiness (d) Surprise (e) Disgust (f) Fear

Figure 2. CGs created by amateur cartoonists that have highest recognition rates.

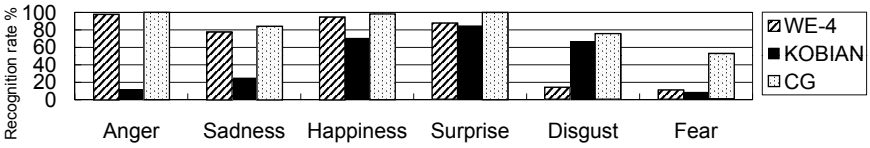


Figure 3. Facial recognition rates of CGs created by amateur cartoonists. This experiment was conducted in Japan in 2010. 58 Subjects (51 men and 7 women, average age: 21.5) participated.

addition, as the number of DoFs increased, the mechanisms of the parts on the face have to be packed. For miniaturization of the head, all the parts on the face were newly developed. The details of these mechanisms are shown below.

2.1 Eyelids

The eyelids consist of the upper eyelid and the lower eyelid. Both sides of the upper eyelid have 1-DoF for opening and closing and 1-DoF for rotating. The lower eyelid has 1-DoF for opening and closing. To obtain the clearance for eye, actuators for these DoFs are kept away from eyes. The opening and closing motion of the upper and lower eyelids is driven by a link mechanism using a DC motor while the rotating motion of the upper eyelid is driven by a wire driven mechanism with an ultrasonic motor TULA-70 (Techno hands Co. Ltd.). The CAD drawing of the mechanism of the eyelid is shown in Fig. 5(a). Detail of mechanism of roll axis of upper lid is shown in Fig. 5(b). As upper lid touches the link mechanism of eye in some conditions,

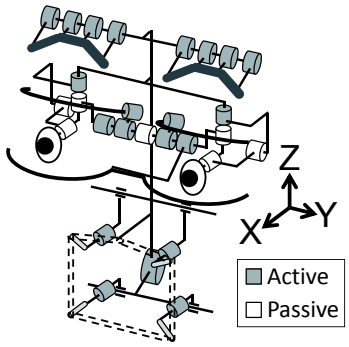
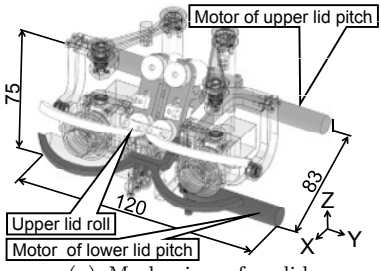


Figure 4. DoF configurations of the face.

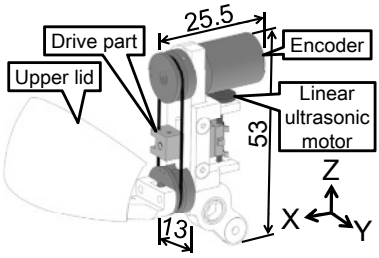
Table 1. DOF configuration

Parts	DoF	Joint
Eye	3	Pitch, Yaw-LR
Upper eyelid	4	Pitch-LR, Roll-LR
Lower eyelid	1	Pitch
Eyebrow	8	Zx4-LR
Upper lip	4	Y-LR, Roll-LR
Lower lip	3	Y, Roll-LR
Jaw	1	Pitch
Sum	24	

* In addition, blue EL sheet for facial color is attached on the forehead.
** LR means that DoF is placed on both left and right side.



(a) Mechanism of eyelids.



(b) Mechanism of upper eyelid roll.

Figure 5. Mechanism of eyelids.

we casted upper lid with urethane rubber.

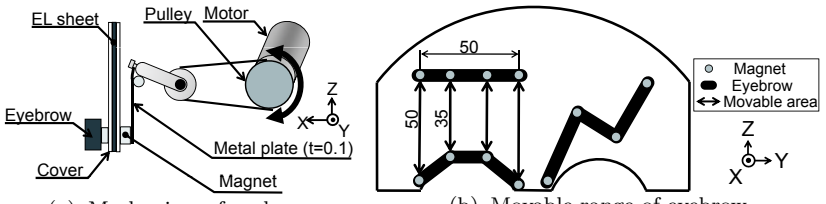
2.2 Eyebrows and Forehead

KOBIAN-R's eyebrows have four equally spaced control points. They move in Z axis and the shape of the eyebrows changes. Similar to the mechanism of the eyebrows of KOBIAN, the eyebrows are driven by the magnet through the head case (N. Endo et al. (2008)). Because of the clearance necessary for upper lid Roll, size of KOBIAN-R's eyebrow mechanism is severely limited in the direction of X axis. Because of that, it was impossible to use the same mechanism as before. A new mechanism using the 0.1[mm] thick metal plate structure has been developed for KOBIAN-R. Fig. 6(a) shows the details of 1-DoF of the eyebrow mechanism. The link with thin metal plate is driven by wire mechanism using DC motor. It pushes and pulls the thin metal plate deforming its shape. Then the magnet on the metal plate moves and the shape of the eyebrows changes. The eyebrows are molded with Septon (KURARAY CO.LTD) and attached magnets to the control points. The movable range of KOBIAN-R's eyebrows is much wider than those of KOBIAN and WE-4 (Fig. 6(b)).

The forehead should change its color when it express "Fear" (Fig. 2(f)) to improve the recognition rate. It was realized by using a blue Electro Luminescence (EL) sheet under the head case.

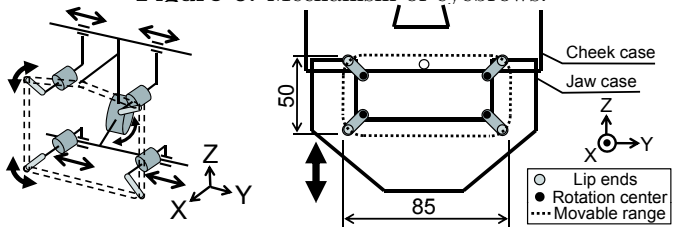
2.3 Mouth

Mouth mechanism is shown in Fig. 7(a). The lip changes its shape pulled from 5 directions. There is one fixed point on the upper face case,



(a) Mechanism of eyebrow. (b) Movable range of eyebrow.

Figure 6. Mechanism of eyebrows.



(a) DoF configuration of mouth. (b) Movable range of mouth.

Figure 7. Mechanism of mouth.

two active control points on the upper lip and two active control points on the lower lip. Active control points can move in Y and Z directions. Mechanisms of lower lip are placed on the jaw structure and driven by the jaw joint. This structure allows the movement of the lip naturally.

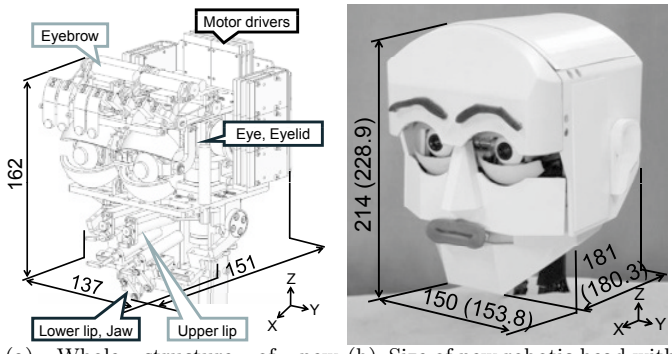
Because lip is stretched four times as large as normal size in some emotions, the lip should be strong and deformed by small power. Threrefore the lip is molded with Septon (KURARAY CO.LTD) and ploughed steel wool is attached in the control points. The movable range of KOBIAN-R's lip is much wider than that of WE-4 and KOBIAN (Fig. 7(b))

2.4 Whole Structure

The overview of the mechanism of the new robotic head is shown in Fig. 8(a). The outer dimension of the head with cover is as large as that of Japanese adult female (Fig. 8(b)). The weight of the head is 1.7 kg. In spite of the number of DoFs, KOBIAN-R's head is much smaller than KOBIAN. Moreover, the weight of KOBIAN-R's head is 0.7 kg lighter than that of KOBIAN. We succeeded reducing in size and weight thanks to the use of customized motor control modules described in T. Otani et al. (2012).

3 Experimental Evaluation

To evaluate the new facial expression ability of KOBIAN-R's head we conducted a questionnaire survay with pictures. 27 subjects (25 men and 2 women; average age was 20.3) participated in the evaluation. The subjects were asked to choose the emotion that they thought the picture of the robot was expressing among a predetermined list ("anger", "happiness", "surprise", "disgust", "sadness", "fear", and "other"). We showed 7 pic-



(a) Whole structure of new robotic head.

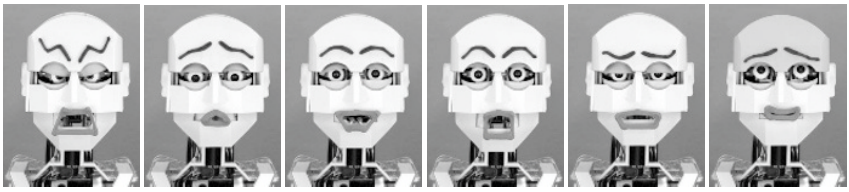
(b) Size of new robotic head with values of average adult Japanese female head in parentheses.

Figure 8. Whole structure of new robotic head.

tures of WE-4RII, KOBIAN, KOBIAN-R and 7 cartoonists' CGs (1 neutral and 6 emotional). Each emotional picture was shown side by side with the picture of the "Neutral" expression as a reference and in random order. The pictures of KOBIAN-R used in the experiment are shown in Fig. 9. The recognition rates of KOBIAN-R's facial expressions are higher than the ones of KOBIAN (Fig. 10). However, the recognition rate of KOBIAN-R for "fear" is 35% lawer than CGs. The difference of recognition rates between CGs and real machine suggests that there are some factors that make impressions of real machine different from CGs. For example, shade or mechanisms can be seen on the real machine. To improve the facial recognition rates and reduce the deference between facial recognition rates of CGs and real machine, the way to tune patterns of facial expressions on the real machine by amateur cartoonists thoughts to be effective.

4 Conclusions and Future Works

To realize robots supporting human life, interacting easier in human environment, communication between robot and human and walking ability are very important. Robots should have the ability to express emotions as well as bipedal walking. However, currently no robots possess both abilities. Since 2007, we have been developing KOBIAN. It is a bipedal walking robot, but since it doesn't have sufficient facial expression ability, we developed a new head that is able to express 6 basic emotions and be sufficiently small and light for bipedal walking. First the authors searched the representative facial expressions. We asked amateur cartoonists to create facial expression CGs and picked the ones with the highest recognition rates for each basic



(a) Anger (b) Sadness (c) Happiness (d) Surprise (e) Disgust (f) Fear

Figure 9. Facial Expression of KOBIAN-R.

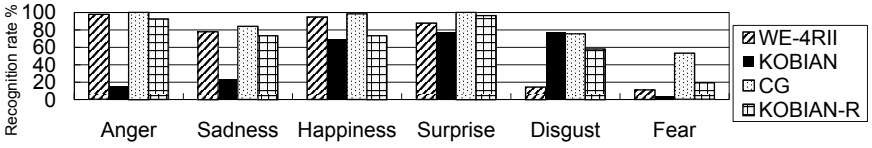


Figure 10. Facial recognition rates of KOBIAN-R. This experiment was conducted in Japan in 2010. 27 Subjects (25 men and 2 women, average age: 20.3) participated.

6 emotions. In order to satisfy size and expressivity requirement, we developed the new head with 24-DoFs, wide movable range of eyelids, eyebrows and mouth and facial color, we designed very compact mechanisms. Questionnaire results confirmed that the new head outperforms the old one in terms of expressions recognition. We developed new head capable of those facial expressions. In future we are planning to adapt the new robotic head on the KOBIAN's body and evaluate the emotional expression with whole body.

Bibliography

Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, *The intelligent ASIMO: System overview and integration*, In *P*roceedings of IEEE/RSJ Int. Conference on Intelligent Robots and Systems, pages 2478–2483, 2002.

K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka and S. Kajita, *Cybernetic Human HRP-4C*, In *P*roceedings of 9th IEEE-RAS International Conference on Humanoid Robots, pages 7–14, 2009.

T. Hashimoto, S. Hitramatsu, T. Tsuji, and H. Kobayashi, *Development of the Face Robot SAYA for Rich Facial Expressions*, In *P*roceedings of SICE-ICASE International Joint Conference, pages 5423–5428, 2006.

H. Ishiguro and S. Nishio, *Building artificial humans to understand humans*, In *J*ournal of Artifical Organs. Vol. 10(3), pages 133–142, 2007.

H. Miwa, et al., *Development of a new human-like head robot WE-4*, In *P*roceedings of IEEE/RSJ International Conference on Intelligent Robots and System, pages 2443–2448, 2002.

K. Itoh, et al., *Mechanical Design of Emotion Expression Humanoid Robot WE-4RII*, In *P*roceedings of 16th CISM-IFToMM Symposium on Robot Design, Dynamics and Control, pages 255–262, 2006.

N. Endo, et al., *Development of Whole-body Emotion Expression Humanoid Robot*, In *P*roceedings of IEEE International Conference on Robotics and Automation, pages 2140–2145, 2008.

Y. Ogura, et al, *Development of a new humanoid robot WABIAN-2*, In *P*roceedings of IEEE International Conference on Robotics and Automation, pages 76–81, 2006.

P. Ekman and W.V. Friesen, *Approach of facial analysis*, Seishin shobo, 1987.

T. Otani, et al., *Development of Distributed Control System and Modularized Motor Controller for Expressive Robotic Head*, In *P*roceedings of 19th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control, 2012.

DESIGN OF A NEW HAND EXOSKELETON FOR REHABILITATION OF POST-STROKE PATIENTS

Mohammad Mozaffari Foumashi* and Marco Troncossi*
and Vincenzo Parenti Castelli*

* Faculty of Engineering, University of Bologna Viale del Risorgimento 2,
Bologna ZIP-Code: 40136, Italy

Abstract This paper presents a novel design of a single degree of freedom planar 12-link mechanism for a finger exoskeleton. The mechanism is sized to each finger of the human hand and attached to the phalanges to control the flexion/extension movements while generating the finger desired grasping trajectory.

1 Introduction

In order to rehabilitate patients with upper limb deficiencies following stroke or peripheral nerve injury, the common practice is based on the thorough and constant execution of proper exercises. In the attempt to improve the neuro-rehabilitation process, many robotic systems that seem to outperform with respect to conventional therapy have been recently proposed (Mehrholz et al., 2008 and Kwakkel et al., 2008). The advantages offered by robot systems are the possibility to automate training exercises (that can prove exhausting for physical therapists), to deliver them with a high repeatability, and to estimate the patient's performance during the rehabilitation period by a number of objective measurements. Many rehabilitation robotic systems have been proposed in the literature.
In this context, the BRAVO project ("Brain computer interfaces for Robotic enhanced Action in Visuo-motOr tasks") is intended to the development of a full upper-limb robotic system for the assistance and rehabilitation of motor impaired users, in sequences of reaches, grasps and manipulations. The novelty proposed by the BRAVO approach is the use of the information arising from the user's attention and intention to control the arm and hand exoskeletons(Bergamasco et al., 2011). Within this framework, the authors are directly involved in the design of a hand exoskeleton, named BHO hand (BRAVO Hand Orthosis), developed with the purpose of recovering

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*,
CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_20,

the patient's ability to control the grasping of cylindrical objects. From the functional point of view, the proposed hand system relies on a basic unit featuring a mechanism with one Degree of Freedom (DOF) designed for guiding each single finger along a proper trajectory. In (Troncossi et al., 2012) two different finger mechanisms have been presented. Both solutions provide advantages and disadvantages as discussed in (Troncossi et al., 2012). Since the first solution suffers from singularities, avoidable at the expenses of a more complex design, and the second solution makes use of prismatic pairs which makes the design of the exoskeleton challenging, in this paper a new finger exoskeleton that overcomes the above said drawbacks is presented. The novel design is a planar 12-link mechanism that, attached to all the human phalanges, control the finger flexion/extension movement. The results of the kinematic and static analyses show the feasibility of the hand exoskeleton, which is now under construction.

2 Feasibility study of a hand exoskeleton

2.1 Technical Specifications:

The rehabilitation protocol targeted by the BRAVO project is focused on the training of patients in the acute phase, with the aim of recovering the basic functions of manipulation. In this context, the BHO hand is conceived to perform a cylindrical grasp. In particular the device should assist the patient both to extend the fingers when approaching the object (against a possible residual spastic force that tends to make the patient keep the hand closed) and to control their motion and force when closing around the object. To this purpose the device will drive both the flexion and extension movements. The thumb is controlled by one actuator (resulting in a coupled motion of the finger phalanges) whereas a second actuator is devoted to the control of the other four fingers (implying the coupling of the single finger phalanges and of the fingers themselves). For a cylindrical grasp, a device with one DOF only would be theoretically sufficient. However, the independent action of the thumb with respect to the four fingers was preferred to properly control the motion coordination in order to correctly grasp the object. All links of the mechanisms are located above the fingers, while the two actuators are placed on a frame fixed to the hand backside.

As for the finger trajectories to be considered as reference for the mechanism synthesis/design, the literature was deeply investigated. Unfortunately, data reported in different papers are mutually inconsistent (likely due to different analysis methodologies). However, since a reference was needed for kinematically representing the task of grasping middle size cylindrical objects (starting from a configuration with fingers being fully extended) the

average trajectory data reported in (Gulke et al., 2010) and relative to the grasping of cylinders with diameters of 55, 70, 90 and 120 mm were considered.
When grasping a cylindrical object phalanges should have a trajectory consistent with the requirement to have all phalanges that come into contact with the cylindrical object to be grasped at the same time (this would avoid to change the pose of the cylinder once it has been grasped). In other word, the motion of the fingers should be synchronous. This condition may be achieved only with a multi degree of freedom hand. Since we have a 1 DOF system for four fingers, the synchronization problem will be solved only with a certain approximation.
As for the forces, the hardest required task is assisting the finger extension due to some possible residual spastic force. Also in this case the literature does not offer definite reference data. The effect of the residual contraction was thus modeled as an external force of 10 N applied at the fingertip of each finger, orthogonal to the third phalanx middle line for every configurations along the finger trajectory. For the opposite movement (i.e. the flexion one), the maximum hand grasping force was set at 30 N, which was considered enough to securely hold small to medium size objects (e.g. a glass full of water). These data are consistent with the experience retrieved from the clinical practice.
As far as the human hand geometry is concerned, average data were taken as reference by combining the anthropometrical proportions retrieved from (Buchholz and Armstrong, 1992) and (Drillis and Contini, 1964) and considering a subject's height of 1.75 m.

2.2 Feasibility study of the proposed architecture:

The basic module of the pursued hand exoskeleton is a planar mechanism attached to one human finger that must replicate the flexion/extension movements of the three anatomical phalanges and makes the motion of the three phalanges be planar (the abduction/adduction of the first phalanx is inhibited). Moreover, a coupled motion of the three phalanges is assumed to be acceptable, thus allowing a 1 DOF mechanism to be used as a driving finger exoskeleton. The driving link of the thumb mechanism must be controlled by one actuator, whereas the driving links of the four mechanisms attached to the four opposite fingers will be connected to the shaft of a second actuator, thus the overall system has two DOFs.
Since all fingers have the same topology, the following section will focus on the synthesis of a one DOF finger mechanism, being the mechanism resized in order to adapt it to the specific finger size it will be coupled with.

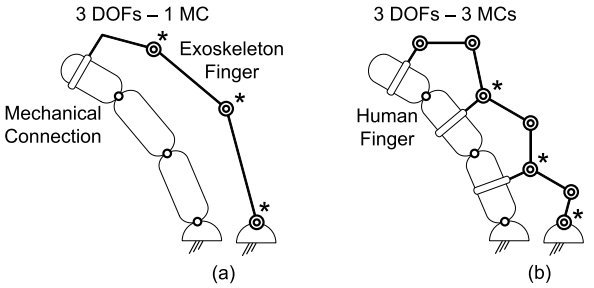


Figure 1. Two finger-exoskeleton design approaches (a) and (b). The star symbol represents actuated joints.

Two different approaches can be adopted to design a finger driving mechanism (Figure 1): (a) to drive the fingertip by a mechanical connection (MC) to the end effector of a 3 DOF mechanism or (b) to drive the finger motion by mechanically connecting all finger phalanges to the exoskeleton. The proposed mechanism relies upon the second approach which includes the phalanges to act as necessary members of the exoskeleton to form a 1 DOF mechanism. It comprises 12 links, also including the three phalanges, interconnected by 16 revolute joints thus making a 1 DOF mechanism. It is worth noting that natural connection between adjacent phalanges is considered as a revolute joint. The mechanism is reported in Figure 2a and its topology is reported in Figure 2b. In particular, the mechanism has the following compositions: 2 quaternary links (links 2 and 3), 4 ternary links (links 6, 8, 9, 11) and 6 binary links (links 1, 4, 5, 7, 10, 12). Figure 2a shows the real mechanism, that comprises three phalanges (1, 2, 3) where the link numbers correspond to those of Figure 2b. For instance, the two phalanges 2 and 3, become the two quaternary links 2 and 3 once the exoskeleton is connected to the finger. The mechanism comprises five loops as shown in Figure 2b. By this mechanism all the desired trajectory requirements and finger interference avoidance with the mechanism links can be satisfied.

The same approach is considered to design a proper mechanism for the thumb. Although from the anatomical point of view the thumb is made up of the same number of phalanges and joints as in the other four fingers, it was decided to fix at a given value the flexion/extension and abduction/adduction movement of the proximal thumb joint CMC(carpometacarpal joint). This assumption was done mainly based on the data obtained from (Gulke et al., 2010). According to this assumption and based on the same mechanism used for the other four fingers (Figure 2b), four links (links

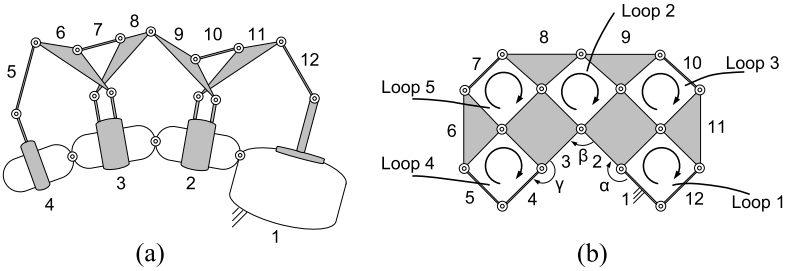


Figure 2. Model (a) and topology (b) of the synthesized 12-link mechanism.

4, 5, 6, 7) of the mechanism are dropped in order to control the lower number of phalanges in the thumb.

3 Design of the new mechanism

After selecting the mechanism topology, the dimensional synthesis was done in order to satisfy the desired constraints. In particular, the finger has to pass through some given configurations. The kinematic and static analyses were also done in order to verify the kinematic constraints and that the forces exerted on the human phalanges at the interface with the mechanism links are in a proper range.

3.1 Dimensional Synthesis

By a trial and error procedure a first acceptable solution was obtained, therefore leaving the adoption of more advanced (sophisticated) techniques ((Erdman, 1981), (McCarthy and Soh, 2010), (Konaka et al., 2006) and (Datta and Deb, 2011) to a further design.

In order to reach the desired trajectory, based on (Gulke et al., 2010) the angles α, β and γ (Figure 2b) in five different configurations from full extension to the minimum grasped cylinder was taken as reference trajectory. Therefore, for each finger mechanism, the synthesis was done to reach these sets of angles in five configurations. The synthesis was done by considering different groups of the mechanism loops in sequence. Namely (Figure 2b): the first group contains loop 1 (links 1, 2, 11, 12); the second group contains loops 2 (links 2, 3, 8, 9) and 3 (links 2, 9, 10, 11); and the third group contains loops 4 (links 5, 6, 3, 4) and 5 (links 6, 7, 8, 3). The synthesis started by focusing on the first group (loop 1). The input of loop 1 (four-bar mechanism) is the orientation of link 12 with respect to link 1 (ground) and the

outputs are the position and orientation of link 2 (proximal phalange) and link 11. The dimensional synthesis was done sequentially for each group in order to reach five desired output angles (α) in five configurations. These configurations started from full extension of the finger and continued to flexion that pass through 4 decreasing diameter of cylindrical object.
The next step is to design the second group which contains two coupled four bar linkages (loops 2 and 3). The inputs of the second group are the positions and the orientations of links 2 and 11, and the outputs are the positions and orientations of link 3 (middle phalange) and 8. The synthesis of the second part (two coupled 4-bar chains) has been done by inspection to reach five desired values for the angle β (the proper angle to grasp the desired cylindrical object). It is worth mentioning that the length of the binary link number 7 plays an important role in controlling the orientation of link 3 (angle β). Finally, the same procedure was done to synthesize the third group (loop 4 and 5) which contain the distal phalange.
The allowed range of motion for the flexion/extension angle of the metacarpophalangeal (α), the proximal interphalangeal (β) and the distal interphalangeal (γ) joints are kept in a proper range: (5÷80), (0÷90) and (0÷20) respectively. Additionally, the link size (mainly the distances between two adjacent joints in one link) are kept in a proper range (10÷52 mm).
The same dimensional synthesis was done for the other finger mechanisms. The synthesis was done based on the lengths of each phalanges of the selected fingers and the proper grasping angles of each joint. The problem of finger synchronization was satisfactorily achieved by minimizing the errors when the group of four fingers grasp a cylindrical object of diameter 60 mm, taken as reference diameter for the selected range (50÷70 mm).
Another constraint to be taken into account is the placement of the links and joints. In the whole range of motion, the links and joints of the mechanism should be kept above the human finger phalanges (above the links 1, 2, 3, 4) not to disturb the movement of the fingers. Of course interference among the links of the mechanism and the phalanges must be avoided in all configurations. Moreover, if two links or joints interfere in their workspace, they have to be manufactured in order to operate in two different planes. The total number of planes for each finger should be as lower as possible.

3.2 Static Analysis

Once the mechanism geometry was defined, the static analysis was performed to calculate the forces exerted by the mechanism links at the interface with the phalanges, and the required driving torque for grasping an object or to balance the possible residual contracting forces.

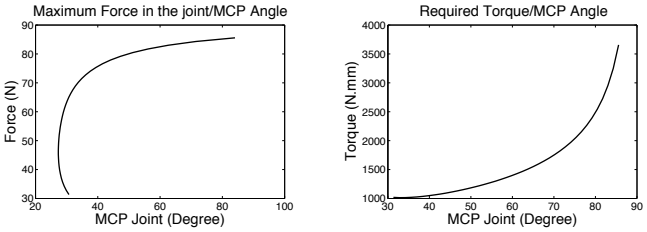


Figure 3. Force between links 1 and 2 and torque between links 1 and 12.

In order to reach this purpose, the static analysis was done to find out the amplitude and the direction of the forces exerted at each joint. These forces were computed as a resultant of a residual contraction modeled as an external force of 10 N applied at the fingertip. The maximum force exerted at joints is about 150 N which occurs in the full extension mode in the joints of the binary link 7. Other forces that should be taken into account are the forces exerted at each human phalanges. In order to calculate these forces, the internal forces in the links 2, 3 and 4 of each fingers are calculated. The maximum force exerted at the finger is about 150 N. It is possible to find the required amount of torque to actuate link (link 5) to balance the force of 10 N in the fingertip. The required torque from full extension to full flexion is increasing from 0.750 Nm to 2 Nm, so the global maximum torque required for group of four fingers would be approximately about 8 Nm. The same static analysis was done for the thumb. In comparison with other four fingers, the maximum required torque takes place in flexed mode (when grasping an object with a force of 30N). The maximum calculated torque was about 3.5 Nm. With reference to Figure 2, Figure 3 shows the force between link number 1 and 2 and the driving torque between link 1 and 12 versus the input angle α for the middle finger.

3.3 Final design

The hand exoskeleton was designed and manufactured and the prototype is currently undergoing bench tests.

4 Conclusion

In the present paper a novel design of a single DOF planar 12-link mechanism is presented. The mechanism is attached to all human phalanges,

which are considered as parts of the kinematic chain. The dimensional synthesis, kinematic and static analyses were done for all fingers separately in order to satisfy the desired constraints. The proposed solution successfully provides the desired trajectory for the specified grasping motion, while the forces exerted on human phalanges at the interface with the mechanism links are in a proper range.

Bibliography

M. Bergamasco, A. Frisoli, M. Fontana, C. Loconsole, D. Leonardis, M. Troncossi, M. Mozaffari-Foumashi, and V. Parenti-Castelli. Preliminary results of bravo project - brain computer interface for robotic enhanced rehabilitation. In *Proc. of the ICORR2011, IEEE 12th Intl Conf on Rehabil. Robotics, Zurich, Switzerland*, 2011.

B. Buchholz and T.J. Armstrong. A kinematic model of the human hand to evaluate its prehensile capabilities. *J. Biomech.*, 25(2):149–162, 1992.

R. Datta and K. Deb. Multi-objective design and analysis of robot gripper configurations using an evolutionary-classical approach. In *Proc. of the 13th conf. on Genetic and evolutionary computation, NY, USA*, 2011.

R. Drillis and R. Contini. Body segment parameters; a survey of measurement techniques. *Artificial limbs*, 25:44–66, 1964.

A. G. Erdman. Three and four precision point kinematic synthesis of planar linkage. *Mechanism and Machine Theory*, 16(3):227–245, 1981.

J. Gulke, NJ. Wachter, T.Geyer, H.Schll, G.Apic, and M. Mentzel. Motion coordination patterns during cylinder grip analyzed with a sensor glove. *J. Hand Surg.AM*, 35(5):797–806, 2010.

A. Konaka, DW. Coitb, and AE. Smithc. Multi-objective optimization using genetic algorithms: A tuorial. *Reliab. Eng. Syst. Safe.*, 91(9):992–1007, 2006.

G. Kwakkel, B. Kollen, and H. Krebs. Effects of robot-assisted therapy on upper limb recovery after stroke: a systematic review. *Neurorehabil Neural Repair*, 22(2):111–121, 2008.

J. M. McCarthy and G. S. Soh. *Geometric Design of Linkages, 2nd ed.* Springer-Verlag, 2010.

J. Mehrholz, T. Platz, J. Kugler, and M. Pohl. Electromechanical and robot-assisted arm training for improving arm function and activities of daily living after stroke. *Cochrane Database Syst. Rev.*, (4):CD006876, 2008.

M. Troncossi, M. Mozaffari Foumashi, M. Carricato, and V. Parenti Castelli. Feasibility study of a hand exoskeleton for rehabilitation of post-stroke patients. In *Proc. ASME 11th Biennial Conf. on Engineering Systems Design and Analysis, ESDA2012, Nantes, France*, 2012.

Guidelines for the design of multi-finger haptic interfaces for the hand

Florian Gosselin[†*]

[†] CEA, LIST, Interactive Robotics Laboratory,
18 route du Panorama, BP6, 92265 Fontenay-aux-Roses, France

Abstract Several novel interaction peripherals recently came to the market. Tactile interfaces and mocap systems deeply changed the way we interact with computers. On the other side, the continuous fall in prices of VR technologies and the development of force feedback haptic interfaces changed the way people develop new products, moving from real to digital mock-ups. Some limitations remain however as no commercially available interface allows natural dexterous interactions in 3D space. Tactile devices are limited to 2D. Data-gloves are difficult to calibrate and do not give any feedback to the user, the later being also true for mocap systems. Haptic devices are equiped with handles and limit the user's dexterity. Multi-finger haptic interfaces are required to overcome these limitations. This paper makes a short review of such devices, gives more details on a two and a three fingers interfaces recently developed for VR applications and gives guidelines for their design.

1 Introduction

Haptic interfaces are used to interact physically with a virtual or distant environment. For optimal efficiency, they must be as transparent as possible. In free space, it calls for a large workspace, low inertia and friction. During interactions, it requires a sufficient force capacity, a large stiffness and a high bandwidth (Massie and Salisbury, 1994; Hayward and Astley, 1996). These requirements are however contradictory and difficult to meet simultaneously, especially when considering complex and dexterous movements.
As a consequence, most commercial haptic interfaces currently available

[*]The author gratefully acknowledge the support of the MUVII Project (IST 2000-28463) funded by the European Commission for the development of the MUVII-IKD interface and the support of the HSI10 Project (I06-510-R) funded by the French government for the development of the IHS10 Glove.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_21,

focus on prop mediated interactions. The prop can be a handle, a stylus or a more specific object. Examples are found in digital design, fitting or VR training systems. The PHANtoM device[1] allows for the digital design of new products using a stylus prop. The Virtuose[2] allows for the manipulation of digital mock-ups via a handle. The LapVR and CathLabVR platforms[3] allow residents to train surgery procedures using surgical intruments. This way the device's design is simplified. However, the interaction is limited to a single tool.

More complex multi-finger haptic interfaces are required to overcome this limitation. Section 2 makes a short review of such devices. Sections 3 and 4 focus more specifically on a two-fingers interface developed for digital handicraft and a three-fingers device intended for the VR design of car cockpits. Finally, section 5 gives guidelines for the design of hand interfaces.

2 Review of existing haptic interfaces for the hand

Haptic interfaces for the hand can be classified in several ways:

- grounded (i.e. fixed on a carrying robot) or wearable (i.e. fixed on the user body, usually the forearm or the hand palm): wearable devices allow for a greater freedom and a (virtually) infinite workspace as they follow the hand movements. However, they tend to be less comfortable as the user has to carry them. The weight of grounded devices is less a problem but the workspace is limited to the carrying robot's one. Moreover, if the hand device is still fixed on the fore-arm or palm, its transparency is also limited, the robot also contributing to the apparent friction and inertia;
- exoskeleton (i.e. fixed on / interacting with different phalanxes) or fingertip device (i.e. interacting only with the tip of each finger): hand exoskeletons theoretically allow any type of interaction, i.e. both precision and power graps (even if almost all hand interfaces concentrate on fingers and do not integrate palm force feedback). However, they tend to be complex and cumbersome. Moreover, they require a long and tedious calibration before use due to a lack of sensors on passive joints (the device's configuration is not fully determined from the measurements of the sole actuators encoders). On the contrary, if properly designed, the configuration of fingertip devices is fully determined from the actuators encoders. They do not require any user specific calibration and are much easier to use. No force is however fed

[1] http://www.sensable.com
[2] http://www.haption.com
[3] http://www.immersion.com

back to the proximal and proximo-distal phalanxes and only precision graps can be simulated efficiently;
- number of fingers: the number of fingers required depends on the application. The thumb and index are sufficient for precise manipulation of small objects. The thumb, index and middle fingers are sufficient for most precision grasps. Thumb, index and coupled (tied together) middle, ring and little fingers allow emulating most precision and power grasp. Five independant fingers are however required for specific tasks, e.g. tactile exploration of surfaces, hand support, ..., at the price of a more complex design and larger size and weight;
- coupled or decoupled kinematics: human phalanxes are coupled and can not be moved independantly. The same couplings can be implemented on haptic interfaces for the hand, with the advantage of a lower number of actuators, limited weigth and size. However, such underactuated devices can not render forces in all directions, even if specific interaction metaphors can partially compensate this limitation;
- fully actuated or partially passive: fully actuated exoskeletons and fingertip devices allow force feedback on each phalanx and in all directions respectively, at the price however of a large number of actuators. To avoid this difficulty, it is possible to actuate the device only in specific directions, usually the closing movement of the fingers. A typical 5 fingers devices has only 5 motors in this case. The design is highly simplified and lighter but the interaction is very limited;
- local or remote actuation: embedded actuation allows for very short and stiff transmissions with limited friction, thus a higher transparency and better force feedback quality. The device is however larger and heavier. On the contrary, remote actuation allows a lighter and slender design but requires long transmissions, usually bowden cables, which introduce friction and limit the bandwidth due to cables' flexibility;
- only force or force and tactile feedback: as all other haptic interfaces, due to the masses of the moving parts, hand devices have a limited bandwidth (typically 0-5Hz to 0-15Hz). The integration of an additional tactile stage under fingertips is a simple way to increase the rendering bandwidth, at the price of a limited added complexity.

All these devices exist in laboratories. However, only one exoskeleton was commercially available: the Cybergrasp (Cyberforce) from Immersion Corporation (Gosselin et al., 2006). This device has 4 fingers and allows force feedback on different phalanxes. It suffers however from several drawbacks which prevented its acceptance and led Immersion to remove it from its catalogue. As an exoskeleton, its structure is dependant on the hand size, thus a long calibration for each user. Moreover, in order to limit the

glove's weight a remote actuation is implemented. Transmission cables introduce friction and flexibility and the force feedback is of medium quality. Finally, the possible applications are very limited as force feedback is only provided in hand opening - closure movements.
Another interesting portable device is the Rutgers Master II (Bouzit et al., 2002). It is composed of 4 small air cylinders placed inside the palm and attached to the fingertips. It is very light and compact. Its configuration does not depend on the fingers' size and the calibration is simplified. However, the feedback still occurs only in hand closure direction. Moreover, the range of finger movements is limited by the minimum size of the air cylinders in closed finger configuration.
To avoid these problems, several fingertip devices were developed, e.g. the Pure Form and HAPTEX hand devices (Frisoli et al., 2007; Fontana et al., 2009) and the Master Finger 2 (Giachritsis et al., 2009). As explained before, these fingertip interfaces benefit from a simplified calibration. They can render forces in all directions on fingertips. Local actuation allows optimal force feedback quality. Each of them however suffers from specific limitations. The Master Finger 2 is grounded and has a limited workspace. The Pure Form device is fixed on the forearm and is relatively large and heavy. In practice, it is mounted on a grounded arm exoskeleton. The Haptex device implements a complex coupled structure to follow the finger shape and is also relatively cumbersome.
This short overview demonstrates that no existing device allows fully dexterous manipulation of a virtual object or a dexterous slave robot with haptic feedback. This is a particularly difficult issue due to specific requirements of hand devices: to allow for a sufficient dexterity, hand devices require a higher number of DOFs than handle devices; hand interfaces are usually used for surface exploration and fine manipulation in a range of low forces were the human hand is particularly sensitive; users usually look at their hand when doing fine manipulation, hence a particulary compact and visually non-intrusive design is required; without using dedicated manipulation metaphors, grasping an object and keeping it in hand requires the continuous application of a sufficient closure force and the device must be dimensionned considering continuous motor torque capacities, contrary to handle devices which usually consider much higher peak torques.
Considering the strength/mass and power/mass ratios of existing materials and actuators, fingertip devices appear as an interesting solution compared to exoskeletons. Their complexity, size and weight is lower while they allow most interactions. Two example fingertip devices will be presented below.

3 A two-fingers haptic interface for digital handicraft

Figure 1 shows the Wearable Haptic Interface for Precise Finger Interaction (WHIPFI). It is a wearable interface fixed on the back of the hand. It is intended for the fine manipulation of small virtual objects during digital handicraft in primary and technical schools. Two fingers are required in this case and the device is composed of only 2 robots, one for the thumb and the second for the index finger. Each robot is composed of a pivot joint, a parallelogram and a 3 DOFs wrist whose axes intersect at the center of the fingertips. The first 3 DOFs are actuated. To allow for the best possible force feedback quality, the actuators are embedded and as short as possible transmissions are used. Finally, small 2 DOFs tactile actuators are integrated on the last links just below the fingertips.

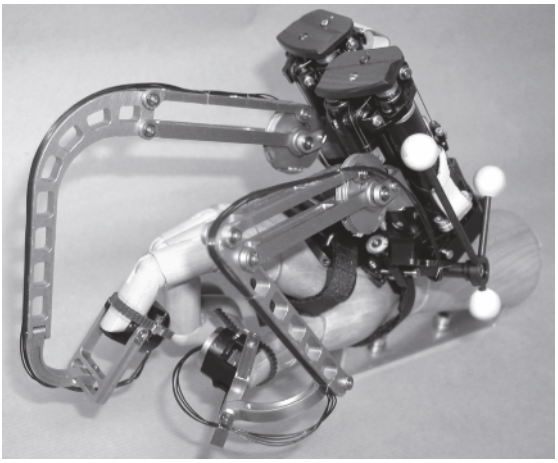


Figure 1. The 2 fingers fingertip haptic device WHIPFI.

The weight of the WHIPFI is 1010g and its workspace is larger than $1m^3$ (it is only limited by the length of the controler cables). The workspace of the robots encompass the finger movements with a (local) position resolution below $60\mu m$, continuous and peak force capacities above $0.8N$ and $4.74N$ and a global stiffness above $900N/m$ (taking into account control and mechanics) in reference position. More details on the device's design and performances can be found in (Gosselin et al., 2005). More details on the educational applications and evalutation results can be found in (Christodoulou et al., 2005).

4 A three-fingers haptic interface for digital mock-up

Figure 2 shows the IHS10 fingertip haptic device. It can be used as a wearable interface or can be fixed on a carrying robot. It is intended for interaction with car cockpits controls. A complete analysis of such interactions shows that most of them are feasible with 3 fingers. Consequently, the device is composed of 3 robots for the thumb, index and middle fingers. Each robot is composed of a pivot joint, a parallelogram and a 2 DOFs wrist whose axes intersect at the center of the fingertips. The last link has a rounded inner shape in order to accomodate small movements around the distal phalanx axis and emulate a 6^{th} virtual axis. All joints are realized with novel large ranges of motion flexible joints, allowing for a much more compact design. Further details will be given in a coming paper. The first 3 DOFs are actuated, remotely to limit the weight and size of the device. To limit the associated friction and flexibility, the transmission is kept as short as possible (the actuators are not grounded but mounted on the carrying robot) and high end transmissions are used (multi articulated aliminium bowden system with inner low friction guides).

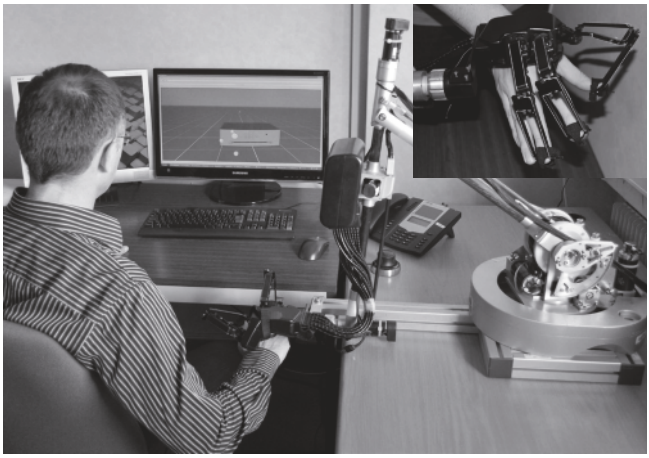


Figure 2. The 3 fingers IHS10 fingertip haptic device.

The weight of the IHS10 device is less than 500g. The workspace of each robot is larger than the fingers' movements. Continuous and peak force capacities are above $4N$ and $10N$ and the control stiffness is above $5000N/m$ in reference position. More details on the device's design and performances will be given in a coming article.

5 Discussion

This section discusses guidelines for the design of hand interfaces.

- size: the IHS10 device is smaller than most other devices. As an example, the WHIPFI accomodates all users, from a small girl (5^{th} percentile, 10 years old) to a big man (95^{th} percentile, 18 years old). Its links are dimensionned accordingly. On the contrary, the IHS10 device makes use of a personalized design. The existing prototype is adapted to a mean man (50^{th} percentile, 18 years old). Thanks to a novel joint design allowing molded mass production, a different size can be proposed for other people. Such design allows a user specific more compact and lighter design. It should be recommended;
- kinematics: the links of most hand devices are cumbersome as they are routed outside the fingers. The IHS10 interface is more compact as the fingers go 'through' the robots' links. Even more compact designs are feasible whith complex redundant structures as proposed in (Nakagawara et al., 2005; Fang et al., 2009);
- structure: contrary to most hand devices which make use of classical ball bearings, the IHS10 device makes use of novel large ranges of motion flexible joints. This solution is more compact and allows for mass production by molding (the links of the frist prototype shown on Figure 2 are still machined from aluminium). A lot of other flexible joints exist in the litterature. We recommend, if possible, this solution;
- actuation: both the PUREFORM, WHIPFI and IHS10 devices make use of backdriveable cable capstan reducers. Other solutions could be used, at the price however of a limited transparency or use of force sensors with associated complexity and reduced reliability, not to mention drift problems. Due to their limitations, embedded actuators and cable capstans do not allow for a sufficient continuous force (necessary to graps objects for a long time). Remote actuation is preferable. However, to avoid extra friction and compliance, the transmission must be kept as short as possible. We recomend this solution only if the device is carried by a robot. For a wearable interface, underactuation seems unavoidable in association with dedicated interaction metaphors.

6 Conclusion and perspectives

This paper presents a short overview of haptic interfaces for the hand. Several distinctive characteristics are listed and different designs are compared with respective advantages and drawbacks. We hope these guidelines will help those interested in designing such devices.

Bibliography

M. Bouzit, G. Burdea, G. Popescu, and R. Boian. The rutgers master ii - new design force feedback glove. *IEEE/ASME Transactions on Mechatronics*, 7(2):256–263, 2002.

S. P. Christodoulou, D. M. Garyfallidou, M. N. Gavala, G. S. Ioannidis, T. S. Papatheodorou, and E. A. Stathi. Haptic devices in virtual reality used for education: designing and educational testing of an innovative system. In *Proceedings Interactive Computer Aided Learning*, 2005.

H. Fang, Z. Xie, and H. Liu. An exoskeleton master hand for controlling dlr/hit hand. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 3703–3708, 2009.

M. Fontana, A. Dettori, F. Salsedo, and M. Bergamasco. Mechanical design of a novel hand exoskeleton for accurate force displaying. In *Proceedings of the International Conference on Robotics and Automation*, pages 1704–1709, 2009.

A. Frisoli, F. Simoncini, M. Bergamasco, and F. Salsedo. Kinematic design of a two contact points haptic interface for the thumb and index fingers of the hand. *Journal of Mechanical Design*, 129:520–529, 2007.

C. Giachritsis, J. Barrio, M. Ferre, A. Wing, and J. Ortego. Evaluation of weight perception during unimanual and bimanual manipulation of virtual objects. In *Proceedings of the International Conference World Haptics*, pages 629–634, 2009.

F. Gosselin, T. Jouan, J. Brisset, and C. Andriot. Design of a wearable haptic interface for precise finger interactions in large virtual environments. In *Proceedings of the International Conference World Haptics*, pages 202–207, 2005.

F. Gosselin, C. Andriot, and P. Fuchs. les dispositifs matriels des interfaces retour deffort. In *Le Trait de la Ralit Virtuelle, 3rd edition, Vol.2 (6)*, pages 135–202. Les Presses de l'Ecole des Mines, 2006.

V. Hayward and O. R. Astley. Performance measures for haptic interfaces. In *Robotics Research: The 7th International Symposium, G. Giralt and G. Hirzinger Eds., Springer Verlag*, pages 195–207, 1996.

T.H. Massie and J.K. Salisbury. The phantom haptic interface : a device for probing virtual objects. In *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 1994.

S. Nakagawara, H. Kajimoto, N. Kawakami, S. Tachi, and I. Kawabuchi. An encounter-type multi-fingered master hand using circuitous joints. In *Proceedings of the International Conference on Robotics and Automation*, pages 2667–2672, 2005.

Development of Anthropomorphic Soft Robotic Hand WSH-1RII

Nobutsuna Endo [*] , Takuya Kojima [*] , Keita Endo [*] , Fumiya Iida [*] , Kenji Hashimoto [*] , Atsuo Takanishi [*†]

[*] Waseda University

[†] Department of Modern Mechanical Engineering, Waseda University; Humanoid Robotics Institute (HRI)

Abstract In the future, we will need service robots capable of natural interaction with people, which often involves the use of hands. In this paper, we aim to evaluate the impression of the interaction between a human and a humanoid robot with anthropomorphic hands. In particular, we focus on the influence of the grip of robot's hands during handshaking. First, we measured pressured places and forces. Based on this result, we selected motors that have enough torque for handshaking. Finally, we developed a new anthropomorphic soft robot hand named WSH-1RII with the soft material structure, enough grip strength, and force sensing. The experimental results confirmed that the hand gives the users a good impression.

1 Introduction

The average age of most industrialized countries is rising fast due to an increased expectation of life and a contemporary reduced child birth rate (OECD, 2004; NIPS, 2003). As an example, in Japan, while today there are about 2.8 workers per retiree, this figure is estimated to fall to 1.4 in 2050. Thus, there is considerable expectation that in the next years there will be a growing need for greater medical and nursing care services. In particular, robots and robot technology (RT)-based assistive devices are expected to play a major role in this elderly society, with an active participation to joint works and community life with humans, as partners and as friends. These robots are expected to be fundamental for helping and assisting elderly and disabled people during their activities of daily living (ADLs), not only by working together with humans, but also by expressing human-like emotions and behaviors.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_22,

As a possible robotic assistant, we developed a new biped humanoid robot, named KOBIAN (Figure 1, Endo et al. (2008)). KOBIAN is capable of human-like movements and of human-like emotion expressions. It is therefore expected to be capable of interacting with surrounding people both physically and psychologically. The hand of course plays a fundamental role during the interaction, which should both be capable of grasping and of gestures.

There are some soft robotic hands from a point of view of prosthetics (Doshi et al., 1998; Carrozza et al., 2005). These soft hands are not designed for robots but for humans. And, these are not evaluated by points of view of psychological effects.

It is not clear that what kind of mechanical configuration of hand is important to ADL-assistive robots, and, particularly, what factors are effective to user's impression. To clarify the affects of robotic hands, we designed and realized the Waseda Soft Hand #1 (WSH-1, Zecca et al. (2006)), and we compared its acceptance with a hard and a mixed (hard/soft) hands. The results showed that the soft hand was preferred to the hard or the mixed ones. Then, we also investigated the most influencing shape parameters of hand which affect on tactile impressions, and developed the Waseda Soft Hand #1 Refined (WSH-1R, Endo et al. (2010)).

However, the grip strength during handshake has not been fully investigated yet. Robotic hand should sense its grip strength in order to adapt to various human hands (big/small, thick/thin, strong/weak, etc) and adjust the strength. WSH-1R is incapable of adaptive handshaking because it does not have enough grip strength and does not have any force sensors. Therefore, we developed a new robotic hand named Waseda Soft Hand #1

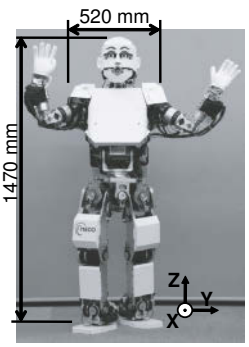


Figure 1. KOBIAN

Refined II (WSH-1RII) which is capable of adaptive handshaking and of adjusting grip strength. In this paper, we present the development of the WSH-1RII and psychological evaluation of grip strength during handshake interaction.

2 Development of Waseda Soft Hand #1 Refined II

There are many factors of robotic hand influencing tactile impression: material, shape, temperature, behavior, and so on. We have evaluated the tactile impression of different material using WSH-1. We evaluated the impression of different shape using WSH-1R. These studies suggested that soft material and rounded shape could contribute to better tactile impression. Our previous soft robotic hand WSH-1R was developed in order to mount on the whole-body emotion expression biped humanoid robot KOBIAN so that KOBIAN could perform interaction with humans. WSH-1R has 4 degrees-of-freedom (DOFs) in each hand, and is capable of adaptable grasping of simple objects (e.g. sphere, cylinder, etc) and is capable of simple gestures (e.g. pointing, peace sign, etc).

In this paper, we aimed at investigating influences of grip strength during handshake interaction on tactile impression. However, WSH-1R does not have enough grip strength, does not have any force sensors, and could not control grip strength.

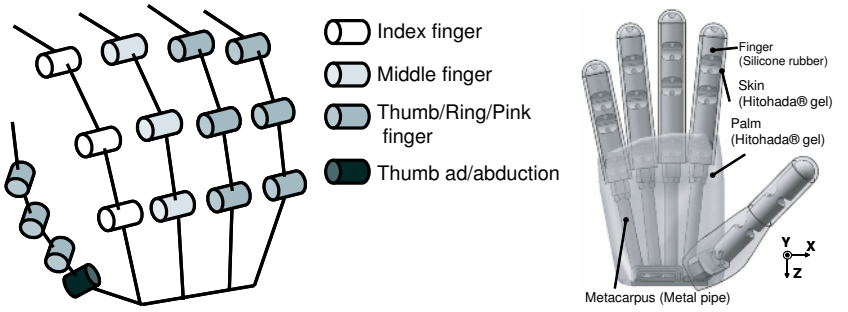
Therefore, we decided to develop a new robotic hand based on the WSH-1R. The specification of the new robotic hand is following:

- Anthropomorphic soft robotic hand
- Capable of grasping and gestures
- Max # of active DOFs: 4 (Figure 2(a))
- Weight ¡ 1.5 [kg] (including forearm)
- Max size: proportional to the height of the robot
- Enough grip strength
- Capable of sensing grip strength

The details of some specifications are described in [4].

Basically, WSH-1R consists of soft material. Each finger is realized as a single peace of silicone (ELASTOSIL M8520, Wacker Asahikasei Silicone Co., Ltd.) and driven by one wire. The wire goes through hollow pipes embedded in the finger pulp. This underactuated mechanism enables to conform to objects of any shape softly and gently without using complicated control. In addition, each finger wears a skin molded by skin-like gel (Hitohada gel, Exseal Co., Ltd.). Figure 3 shows a cross-section view of the finger. The palm is molded by Hitohada gel. Hollow pipes are inside of

palm and mimicking the metacarpal bones of the human hand. The wires for driving fingers also go through the hollow pipes. Figure 2(b) shows a CAD image of the whole hand structure.



(a) DoF configuration. (b) Structure of the hand.

Figure 2. DoF and structure of WSH-1RII.

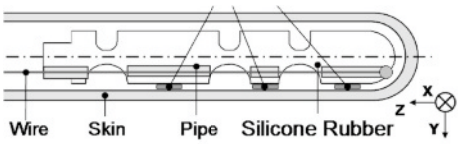


Figure 3. Structure of the finger.

We chose motors, gear ratio, and mechanism for the new robotic hand. We calculated required tension of each finger by following expression:

$$T \cdot l_t = (F_1 \cdot l_1 + F_2 \cdot l_2 + F_3 \cdot l_3) + (M_1 + M2 + M3)$$

where F_n: forces on finger pulps, l_n: distances between force point and knuckle, M_n: moments by elasticity of each knuckle, T: wire tension, l_t: distance between central axis of finger and wire (Figure 4).

Then, we calculated required torque of each DOF and chose motors and gear ratio. On the thumb, the ring, the little fingers, required tensions are 37[N], 60[N], and 57[N]. The distribution ratio of the equalizer mechanism in WSH-1 was 2:1:1. So, we used a variable speed pulley and improved the tension equalizer mechanism as shown in Figure 5.

We mounted 14 sheet force sensors (FlexiForce, Nitta, Co., Ltd.) for measurement of grip strength of the robotic hand. The sensors were mounted on the finger pulps and were covered with skins molded by Hitohada gel (Figure 3). We implemented force control of each motor.

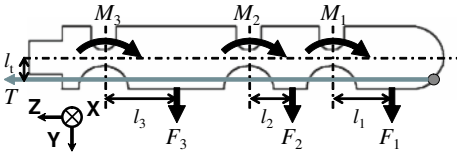
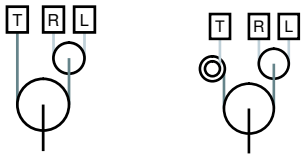


Figure 4. Tension calculation.



(a) WSH-1R. (b) WSH-1RII.

Figure 5. Tension equalizer of WSH-1R and WSH-RII.

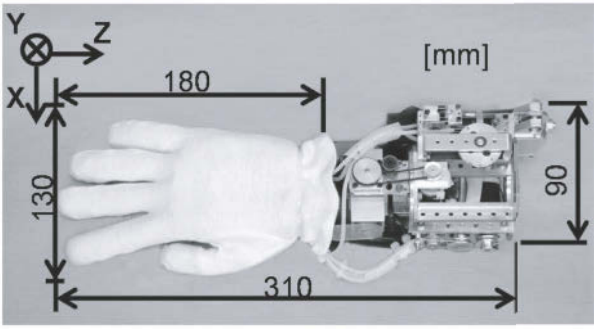


Figure 6. Waseda Soft Hand #1 Refined II.

Figure 6 shows the new anthropomorphic soft robotic hand Waseda Soft Hand #1 Refined II (WSH-1RII). The weight of hand and forearm is 1.3 [kg].

3 Experimental Evaluation of Tactile Impression by Difference in Grip Strength

We investigated effects of grip strength on tactile impression during hand-shake interaction. 21 subjects (18 men, 3 women, average age: 23) were asked to shake hands with human hand, WSH-1R, and WSH-1RII blindly

and to answer 16 semantic differential questions shown in Table 1. A 7-points Likert scale from -3 to +3 was used, with 0 meaning the neutral response. The human hand was the experimenter's one (man, 23 years old). The subjects were asked to perform normal handshake with the hands. In the handshake with WSH-1R, the WSH-1R did not grip. In the handshake with WSH-1RII, the handshake trials were conducted on 5 scale of grip strength (from zero to maximum). Totally, each subject did 7 handshake interactions in random order.

Table 1. Semantic differential questions.

#	Semantic differential pairs
1	Hard / Soft
2	Warm / Cold
3	Desirable / Undesirable
4	Pleasant / Unpleasant
5	Safe / Dangerous
6	Positive feeling / Negative feeling
7	Human-like / Machine-like
8	Smooth / Rough
9	Enjoyable to touch / Painful to touch
10	Gentle / Frightening
11	Like / Dislike
12	Familiar / Unfamiliar
13	Positive / Negative
14	Positive sensation / Negative sensation
15	Friendly / Unfriendly
16	Strong / Weak

The result is shown in Figure 7.

It is clearly shown that changes in the grip strength affect on tactile impressions. The score of 1) hard/soft is increased relative to the grip strength. On the contrary, the score of 5) safe/dangerous is decreased. And, there are peaks in the other scores. This suggests that there is adequate value of grip strength about impression which robotic hand give human.

Overall, however, the scores of WSH-1RII are inferior to those of human hand. So, we have to identify the other factors influencing tactile impression and develop a better robotic hand.

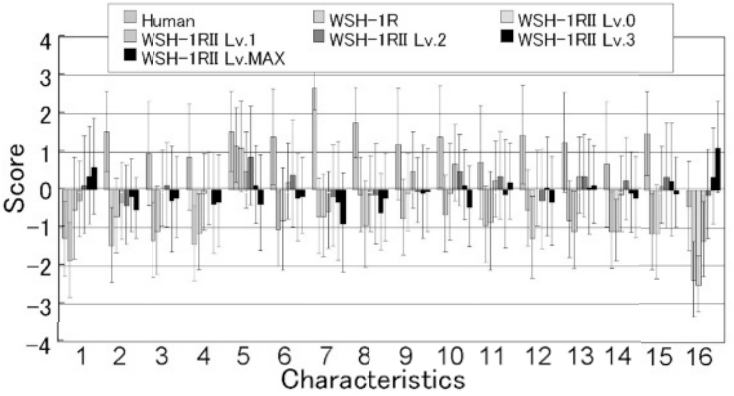


Figure 7. Comparison in different hand and different grip strength.

4 Conclusion and Future Works

In a society that is getting older year by year, there is a considerable expectation that robot technology could provide a substantial help to the ageing population. In order to achieve a smooth and natural integration of the robots, emotion expression plays a major role. A key aspect in successfully emotion expression is the presence of a human-like hand, capable not only of grasping but also of signs and gestures. The authors have been developed anthropomorphic soft robotic hand based on this perspective.

In this paper, we presented the development of the new anthropomorphic soft robotic hand WSH-1RII. The degree of freedom in each hand is 4-DOFs. Each finger is modeled by soft material and underactuated by one wire. The palm is mainly modeled by soft material and has the human-like structure. Adaptable grasping is realized by the tension-equalizer mechanism. Additionally, the hand is capable of grip strength control.

And we also presented the evaluation of the effects of grip strength during handshake interaction. The preliminary experimental results show that the tactile impressions can change by grip strength and some impressions have adequate grip strength.

In the next future, we will investigate the other factors influencing tactile impression such as temperature, gripping behavior, and so on. Besides, a questionnaire is a subjective evaluation method and reflects just total impression. So, We should establish an evaluation method by measuring mental status of human objectively, quantitatively, and in real-time in order to clarify effects of factors of hand and its behavior.

Acknowledgment

This work is supported by the WABOT-HOUSE Project of Gifu Prefecture and the New Energy and Industrial Technology Development Organization (NEDO), and was conducted by Humanoid Robotics Institute (HRI) at Waseda University. This work was supported in part by Global COE Program "Global Robot Academia" from the Ministry of Education, Culture, Sports, Science and Technology of Japan. KOBIAN has been designed by 3D CAD software "SolidWorks". Special thanks to SolidWorks Japan K.K. for the software contribution.

Bibliography

Organization for Economic Cooperation and Development *Population statistics*, 2004.

National Institute of Population and Social Security Research *Population statistics of Japan 2003*, 2003.

Nobutsuna Endo, Shimpei Momoki, Massimiliano Zecca, Minoru Saito, Yu Mizoguchi, Kazuko Itoh, and Atsuo Takanishi *Development of Whole-body Emotion Expression Humanoid Robot*, In *P*roceedings of the 2008 IEEE International Conference on Robotics and Automation, pages 2140–2145, 2008.

Rajiv Doshi, Clement Yeh, Maurice LeBlanc, *The design and development of a gloveless endoskeletal prosthetic hand*, In *J*ournal of Rehabilitation Research and Development Vol. 35 No. 4, pages 388–395, 1998.

Maria Chiara Carrozza, Giovanni Cappiello, Giovanni Stellin, Franco Zaccone, Fabrizio Vecchi, Silvestro Micera, Paolo Dario *A Cosmetic Prosthetic Hand with Tendon Driven Under-Actuated Mechanism and Compliant Joints: Ongoing Research and Preliminary Results*, In *P*roceedings of the 2005 IEEE International Conference on Robotics and Automation, pages 2661–2666, 2005.

M. Zecca, K. Endo, N. Endo, Y. Mizoguchi, T. Kusano, K. Itoh, and A. Takanishi *Design and Evaluation of The Soft Hand WSH-1 For The Emotion Expression Humanoid Robot KOBIAN*, In *P*roceedings of the 6th International Conference of the International Society for Gerontechnology, 2008.

Nobutsuna Endo, Fumiya Iida, Keita Endo, Yu Mizoguchi, Massimiliano Zecca, and Atsuo Takanishi *Development of the Anthropomorphic Soft Robotic Hand WSH-1R*, In *P*roceedings of the First IFToMM Asian Conference on Mechanism and Machine Science (Asian-MMS 2010), 250162, 2010.

Development of Distributed Control System and Modularized Motor Controller for Expressive Robotic Head

Takuya Otani [*] , Tatsuhiro Kishi [*] , Przemyslaw Kryczka [*] , Nobutsuna Endo [*] , Kenji Hashimoto [*†] and Atsuo Takanishi [†‡]

[*] Graduate School of Science and Engineering, Waseda University, Tokyo, Japan
[†] Department of Modern Mechanical Engineering, Waseda University, Japan
[‡] Humanoid Robotics Institute (HRI), Waseda University, Japan

Abstract This paper describes the development of a distributed control system and a small motor controller unit modularized to a controller module and motor driver modules for a humanoid robot. The humanoid robot, KOBIAN, had a centralized control system. It was difficult to increase the number of motors because of PC's I/O channel limit. We installed the distributed control system in the new head to increase the number of motors. Moreover, the modularized motor controller is developed to use in the various parts of the body. The small flexible motor controller is composed of a controller module and motor driver modules. Therefore, changing motor driver modules allows control of various components with the same controller module. This small motor controller unit can control 4 DC motors and read 8 channel of analog sensory data. We evaluated the performance of the developed motor controller and controlled the new robot's head.

1 INTRODUCTION

We have developed a whole body emotional expression biped humanoid robot named KOBIAN (N. Endo, et al. (2010)) (see Fig. 1). Its height was 1400 mm and it had 48 degrees of freedom. However, it had a centralized control system, in which all actuators and sensors are connected to a central controller. We must rewire to add motors or sensors. Moreover, that system has low expandability because the number of motors and sensors is limited by the number of I/O channels of a central controller. On the other hand, a distributed control system is composed of distributed controller units. The controller units control motors and read data from sensors located in close

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_23,

proximity. They cooperate with each other and control the whole robot's body. It reduces the amount of cables to a central controller and enhances expandability because it is possible to increase I/O or A/D channels easily. KOBIAN's head was based on emotion expression humanoid robot, WE-4. WE-4's head has 22 motors and various sensors. On the contrary, KOBIAN's head decreased to 7 motors and no sensors because of PC's I/O channels limit. One solution could be increasing the number of PC's I/O channels by adding another interface board, but it also sized up KOBIAN's backpack. A big backpack affects human-like appearance, in the end we decreased the number of motors. However, we need to increase new motors to improve facial expression. Therefore, we develop new head controlled by a distributed control system to increase the number of motors. The new head (KOBIAN-R's head (T. Kishi, et al. (2012))) has 24 motors and various sensors. Furthermore, its size should be equal to average of adult female's head size (153.8 x 228.9 x 180.3 mm). Therefore, there is no much space for electronic units in this head. The motor controller should be divided to be smaller to implement between head's parts and outer covers of the head. Since, the motor controller controlling single motor would result in a big system, the motor controller should control 4 DC motors and read 4 sensors. 7 motor controllers can control all motors and read all sensors.

Various humanoid robots controlled by a distributed control system have been developed. Their system architecture and motor controllers are specific to each robot. ASIMO (M. Hirose and K. Ogawa (2006)) has CPUs used in each of the robot's several sub-systems, including the actuation of its arms and legs, audio-visual sensing and recognition, communication with the operator and power management. HRP-4C (K. Kaneko, et al. (2009)) has a central PC and 2 kinds of motor controllers. Ito et al. developed the small motor driver integrating sensor circuits. It is composed of motor drivers and a control board handling them.

These humanoid robots have various components such as motors, sensors and some kinds of controller units to control these components. However, controller units are not customized to each segment of the robot, because suiting all of them is inefficient. Therefore, we consider that a controller unit should be divided into the controller module and the function module to use the controller module in various segments of the robot. According to it, we can develop the suitable controller unit by only changing the function module. This way, we can develop the new module without wasting time for debugging of the communication and control sub-systems.

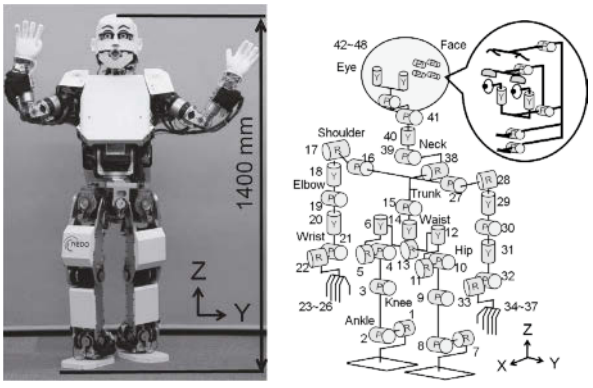
In this paper, we present the development of the distributed control system and the motor controller modularized to the controller module and the motor driver module as a function module for KOBIAN-R's (KOBIAN -

Refined) head. Changing function modules allows control of various components with the same controller module. We evaluated the performance of the developed motor controller. Moreover, we incorporated the motor controllers to KOBIAN-R's head and controlled it.

2 DEVELOPMENT OF MOTOR CONTROLLER

2.1 Modular Architecture

We divided the controller unit into the controller module and the function module. When we need other functions, we change function modules or develop a new one. However, the structure of stacking boards with a pair of board-to-board connectors has two problems. The first problem is maximum length of the big function module is too long to mount in small space. Another problem is that it is not possible to change a part of the module's function. Therefore, the controller module has 2 board-to-board connectors on top and bottom side. Almost all of microcontroller's pins are connected to those connectors. However, dividing pins into 2 connectors make it impossible to use the big function module needed all microcontroller's function. To solve it, 2 connectors' pin configuration is same and symmetrical and 2 connectors connect to each other(see Fig. 2). Thanks to this solution, the function module can be divided into two boards and the same divided function modules can be connected with both sides of the controller module. It decreases maximum length of the unit, and enables us to make the big function module to decrease maximum height. Furthermore, divided modules can be changed to other function modules. Divided function modules are easy to suit various requirements.



(a) Overview (b) DOF configuration

Figure 1. Emotional expression humanoid robot KOBIAN.

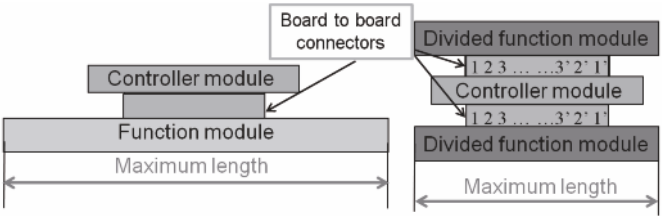


Figure 2. Stacking module.
The number in the connectors means microcontrollers functional pin and apostrophe number means different pin but has same function. 2 connectors shared these pins and its configuration is symmetrical.

2.2 Controller Module

The developed system has a master-slave communication architecture and is controlled by the main on-board PC. Each motor controller handles the local motor control loop and data acquisition of analog sensors. The PC calculates the whole body motion and sends the reference data of motors to motor controllers. The network topology is important to wiring size. Star topology consists of a central node connected with all other nodes. Therefore, the amount of cables is enormous. To reduce the cables, we choose line topology as a communication interface RS-485 because its high resistance to noises. The CPU of the controller module is a STMicroelectronics 32-bit microcontroller STM32F103VG. It has a lot of functions, for example several Analog-to-Digital converters, 6 encoder counters and integrated USART interfaces. Moreover, the controller module has a voltage regulator to stabilize microcontroller's power and a transceiver IC for RS-485.

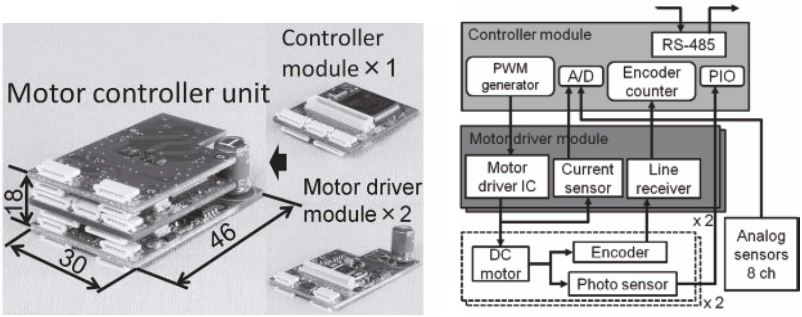
2.3 Motor Driver Module

The motor driver module is one of the function modules. It is equipped with STMicroelectronics H-bridges IC L6227Q, a hall-effect based current sensor for measuring the motor current, a line receiver for incremental encoders, and 2 connectors for photo sensors to initialize the encoder position. This motor driver can control PWM signals for 2 actuators. PWM frequency is up to 100 kHz.

2.4 Motor Controller Unit

The motor controller unit (Fig. 3 (a)) is composed of the controller module and 2 motor driver modules. It can control 4 DC motors and read 8 ch of analog sensory data. The schematic view is presented in Fig. 3 (b) and the specification is in Table 1. Comparing with other motor controllers, the

developed motor controller can read ambiguous sensors and control current, velocity and position loop. Moreover, its size is very small.



(a) Overview (b) Schematic view

Figure 3. Motor controller unit.

Table 1. Specifications of the developed and relevant motor controllers.

	Developed motor controller	4-Axes DC servo (Hibot (2008))	H8-18 FPGA-01D x 4 (N. Ito, et al. (2011))
Motors	4	4	4
Continuous current A	0.7	5	7
Sensors ch	4 Encoders, 4 Photo sensors, 4 Current sensors, 8 A/D	2 Encoders, 8 A/D	1 Force sensor, 1 Temperature, 1 Current sensor, 3 Voltage sensors, 2 A/D
Control modes	Current,Velocity and Position	Velocity, Position	Force, Position
Size mm	46 x 30 x 18	80 x 59 x 8.2	26 x 28, (26 x 46) x 4
Composition	Controller module, Motor driver module	Motor controller	Controller board, Motor driver x 4

3 EXPERIMENTS

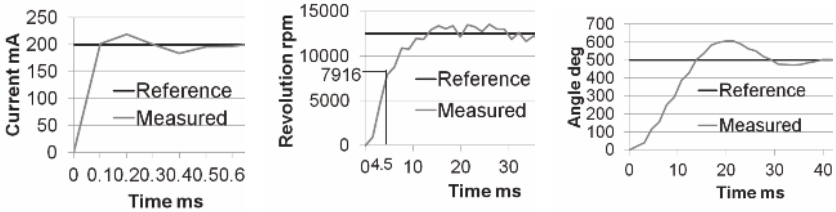
3.1 Communication with PC

We evaluated the reliability of the communication between the PC and the motor controller. We investigated the number of communication errors in million trials during which the PC sent the message which request returning the number to the motor controller and it returned the number. The communication baud rate was 115 kbps limited by the PC in this experiment. Cable length between the PC and the motor controller was 2.5

m. As a result, the error did not occur. Communication's reliability was confirmed.

3.2 Current, Velocity and Position Control Experiments

We investigated the step response for evaluating the current, velocity and position control's response time. We used the brushed DC motor RE10 (1.5 W by maxon Co. Ltd.), and experimented in no load operating condition with the rated voltage 24 V. Current control frequency was 10 kHz, and velocity control and position control frequency were 1 kHz. PID gain of each control modes were adjusted to decrease the response time. According to the motor specification, the number of revolutions when mechanical time constant (4.53 ms) passed from the time when the rated voltage is applied to the motor is 7815 rpm. In velocity control and position control, we compared the number of revolutions after mechanical time constant with this specification data. The experimental results are presented in Fig. 4. In current control, the rise time was 0.1 ms. In velocity control, the result shows that the number of revolutions was 7916 rpm at 4.5 ms. Moreover, in position control, the number of revolutions was 8089 rpm. The results fulfill the motor specification. As a result, the motor controller can fully utilize the motor's performance.



(a) Current control (b) Velocity control (c) Position control

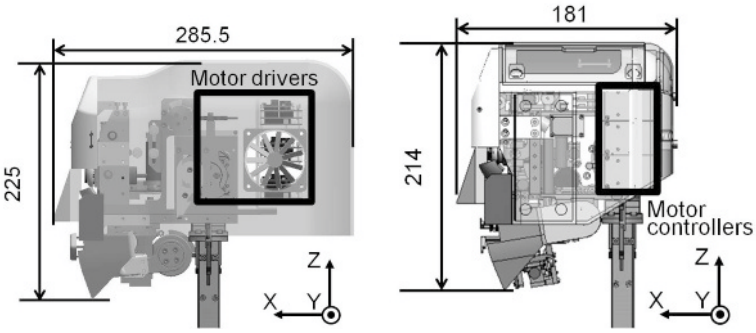
Figure 4. Step responses.

3.3 Robotic implementation

We evaluated various functional performance of the motor controller by incorporation to KOBIAN-R's head and control of it. We describe the distributed control system on the KOBIAN-R's head. Each motor controller has its identification number to be identified from the PC. First, the PC checks which motor controller units are connected. Second, the PC sends the configuration data, such as the information of connected motors, motor's position limit and PID control gain to each controller module. After that, the PC starts control of the head by setting the motor controller to a control

mode. The PC calculates the facial pattern and sends the control reference data to motor controllers. Motor controllers handle the motor control loop and sensor acquisition. Only when the PC sends the command to return the data, the motor controller returns.

KOBIAN's head (Fig. 5(a)) could perform different facial expressions with only 7-DOFs. However, the head had 7 big motor drivers (70 x 42 x 30 mm) and measured 172 x 225 x 285.5 mm. It was bigger than the average size of adult female's head. On the other hand, the 7 motor controllers control 22 DC motors, 2 ultrasonic motors, 1 electroluminescence sheet, 14 tactile sensors and 3 olfactory sensors in the newly developed KOBIAN-R's head (Fig. 5(b)). Thanks to a lot of motor controller's AD channels, it can distinguish the smells of alcohol, ammonia and cigarette smoke with olfactory sensors. And, the head can also recognize the difference in touching behaviors such as "push", "hit" and "stroke". Moreover, the size is 150 x 214 x 181 mm. Thanks to the small motor controllers, it is equal to adult female's head size.



(a) Old head (b) New head

Figure 5. KOBIAN's head.

4 CONCLUSIONS AND FUTURE WORKS

In this paper, a distributed control system of the humanoid robot's head and the motor controller composed of the controller module and the motor driver modules are presented. Changing motor driver modules allows control of various components and decreasing debugging time on the communication and control sub-systems. The controller module has 2 board-to-board connectors to connect with function modules. Therefore, the function module can be divided into 2 modules, and the same divided function modules can be connected with both sides of the controller module. Thanks to this solution, the controller unit can be easily changed to suit requirements. This

motor controller can control 4 brushed DC or ultrasonic motors and read 8 channels extra analog sensors, measuring only 30 x 46 x 18 mm. It is very small considering its various functions. We evaluated the performance of the developed motor controller. Moreover, we confirmed 7 motor controllers can control KOBIAN-R's head, including 22 DC motors, 2 ultrasonic motors, 1 EL sheet, 14 tactile sensors and 3 olfactory sensors. Furthermore, KOBIAN-R's head is equal to adult female's head size because the motor controllers can control ambiguous components and it is very small. In the near future, we plan to develop some function modules and new humanoid robot controlling its whole body by this controller module.

Acknowledgements

This study was conducted as part of the Research Institute for Science and Engineering, Waseda University, and as part of the humanoid project at the Humanoid Robotics Institute, Waseda University. It was also supported in part by RoboSoM project from the European FP7 program (Grant agreement No. 248366), Global COE Program "Global Robot Academia" from the Ministry of Education, Culture, Sports, Science and Technology of Japan, Grant-in-Aid for Research Activity Start-up (22860061), STMicroelectoronics, SolidWorks Japan K.K., and DYDEN Corporation, whom we thank for their financial and technical support.

Bibliography

N. Endo, et al., *Modular Design of Emotion Expression Humanoid Robot KOBIAN*, In *R*OMANSY18, Proc. of the 18th CISM-IFToMM Symposium on Robot Design, Dynamics and Control, pages 465–472, 2010.

T. Kishi, et al. *Development of expressive robotic head for humanoid robot*, In *R*OMANSY19, Proc. of the 19th CISM-IFToMM Symposium on Robot Design, Dynamics and Control, 2012, to be published.

M. Hirose and K. Ogawa, *Honda humanoid robots development*, In *J*ournal of The Royal Society, pages 11–19, 2006.

K. Kaneko, et al., *Cybernetic Human HRP-4C*, In *P*roc. of the 9th IEEE-RAS International Conference on Humanoid Robots, pages 7–14, 2009.

N. Ito, et al., *Development of Small Motor Driver Integrating Sensor Circuit and Interchangeable Communication Board*, In *J*ournal of Robotics and Mechatronics, pages 443–450, 2011.

Hibot Corporation Webpage, http://www.hibot.co.jp/top.php?lang=en.

Chapter IV
Robot Control

Null-Space Impedance Control for Physical Human-Robot Interaction

Luigi Villani,[*] Hamid Sadeghian,[†] and Bruno Siciliano[*]

[*] Dipartimento di Informatica e Sistemistica,
Università di Napoli Federico II, Italy
[†] Mechanical Engineering Department,
Isfahan University of Technology, Iran

Abstract In this paper two approaches for the correct task execution during null-space impedance control of a kinematically redundant robot are presented. The algorithms guarantee safe and dependable reaction of the robot during deliberate or accidental interaction with the environment, thanks to null-space impedance control. Moreover, the correct execution of the task assigned to the end-effector is ensured by control laws relying on two different observers. One is based on task space information and the other on the generalized momentum of the robot. The performance of the proposed control is verified through numerical simulations on 7R KUKA lightweight robot arm.

1 Introduction

New applications where robots work near humans are growing rapidly. Unlike the industrial robots which are stiff with high impedance, the robots working with humans must be designed with high degree of compliance and safety. In these applications, not only unexpected impacts of robots with humans are likely to happen, but also intentional physical Human-Robot Interaction (pHRI) may be required (De Santis et al., 2008). To cope with these situations, different strategies are possible. For example, the manipulator can be covered with a sensitive skin (Lumelsky and Cheung, 2001) capable of measuring the interaction forces. Alternatively, these forces can be estimated from joint positions or torques by means of suitable observers (Haddadin et al., 2008). In any case, suitable control strategies must be adopted to increase robot's compliance (Yoshikawa and Khatib, 2008). Passive compliance can be also introduced by using elastic decoupling between the actuator and the driven link (Bicchi and Tonietti, 2004).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*,
CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_24,

Impedance control represents a very suitable framework for controlling robots in contact with an unknown environment. The problem of impedance control has been extensively studied in the literature. The compliant behavior usually is realized in the task space to control the interaction of the end-effector (Ott, 2008). However, an impedance behavior can be imposed also in the joint space to ensure safety.

A complete theoretical and empirical evaluation of different operational space control techniques for redundant manipulators has been investigated by Nakanishi et al. (2008). One approach to deal with this redundant degrees of freedom is multi-priority control, that can be performed both in kinematic (Siciliano and Slotine, 1991) and dynamic level (Khatib et al., 2004).

De Luca and Ferrajoli (2008) presented a method for fast collision detection and safe reaction based on generalized momentum of the robot, without using any torque sensor. The redundancy of the robot was exploited to preserve as much as possible the execution of the end-effector task by projecting the reaction torques into the null space of the main task.

This paper investigates the null-space impedance control previously presented in (Sadeghian et al., 2011) for robots that do not have joint torque sensors. A new nonlinear control scheme is introduced to estimate the applied external torque on the body of the robot by using a suitable task-based observer. Asymptotic convergence of the main task tracking error is ensured, while at the same time, the joint space impedance is realized in the null space of the main task. Furthermore, the same goal also is pursued by means of a momentum based observer. An example of the application scenario is depicted in Fig. 1, where a redundant robot working on a table experiences a contact with a human. This contact may produce error on the main task of the robot if active compliance is used to ensure safe interaction. Our approach allows to minimize the error in the task space while ensuring compliance in the null space.

2 Null-Space Impedance Control

Dynamic model of a n-link robot manipulator can be expressed by

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{\tau}_{ext} = \boldsymbol{\tau}, \tag{1}$$

with standard notation. In this model, $\boldsymbol{\tau}$ is the input vector torque while $\boldsymbol{\tau}_{ext}$ is the torque resulting from external interaction. If the manipulator is equipped with torque sensors in its joints or force sensors on the interaction points, the external torque can be measured or estimated. In this paper, we assume that these sensors are not available. The well-known model-based

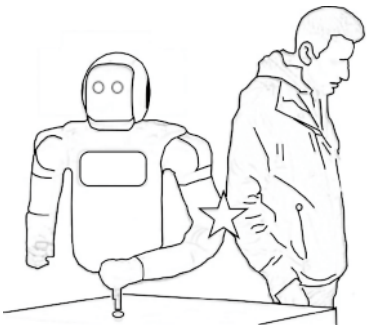


Figure 1. Robot in interaction with a human.

resolved acceleration control can be adopted to compute the driving torques

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_c + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}), \tag{2}$$

where $\ddot{\boldsymbol{q}}_c$ is the joint command acceleration to be suitably designed.

For a redundant manipulator, redundancy lets us to have some kind of joint impedance and task space control simultaneously. The so-called null-space impedance can be realized in the null space of the main task to control the interaction on the robot's body. The corresponding command joint acceleration in (2) is given by

$$\ddot{\boldsymbol{q}}_c = \boldsymbol{J}^{\dagger}(\ddot{\boldsymbol{x}}_c - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) + \boldsymbol{N}(\ddot{\boldsymbol{q}}_d + \boldsymbol{M}_d^{-1}(\boldsymbol{B}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_d\tilde{\boldsymbol{q}})), \tag{3}$$

which produces the task space and null-space closed-loop behavior respectively as follows

$$\begin{aligned} &\ddot{\boldsymbol{x}}_c - \ddot{\boldsymbol{x}} = \boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{\tau}_{ext}, \\ &\boldsymbol{N}(\ddot{\tilde{\boldsymbol{q}}} + \boldsymbol{M}_d^{-1}(\boldsymbol{B}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_d\tilde{\boldsymbol{q}}) - \boldsymbol{M}^{-1}\boldsymbol{\tau}_{ext}) = \boldsymbol{0}. \end{aligned} \tag{4}$$

Here $\ddot{\boldsymbol{x}}_c$ is the command acceleration in the task space, $\tilde{\boldsymbol{q}} = \boldsymbol{q}_d - \boldsymbol{q}$ where $\boldsymbol{q}_d$ is the desired trajectory or a rest configuration in the joint space, $\boldsymbol{J}^{\dagger}$ is any (weighted) right pseudo-inverse of the task space Jacobian matrix $\boldsymbol{J}$, $\boldsymbol{N} = (\boldsymbol{I} - \boldsymbol{J}^{\dagger}\boldsymbol{J})$ is the null-space projection matrix, and $\boldsymbol{M}_d, \boldsymbol{B}_d$ and $\boldsymbol{K}_d$ are the impedance matrices. This choice of $\ddot{\boldsymbol{q}}_c$ allows the joint space impedance in the null-space of the main task to be realized, provided that the desired inertia matrix is chosen as $\boldsymbol{M}_d = \boldsymbol{M}(\boldsymbol{q})$. On the other hand, the main task experiences errors as a result of the external torques that are applied on the robot's body. It can be easily shown that an arbitrary $\boldsymbol{M}_d$

can be set only if the measurement of $\boldsymbol{\tau}_{ext}$ is available to the controller; in this case, also the influence of $\boldsymbol{\tau}_{ext}$ on the main task can be cancelled out.

3 Task-Based Observer

The following theorem is given for correct execution of the main task, while ensuring a compliant behaviour of the robot's body.

Theorem: Let us denote with $\hat{\boldsymbol{\tau}}$ the estimated external torque and with $\tilde{\boldsymbol{\tau}} = \boldsymbol{\tau}_{ext} - \hat{\boldsymbol{\tau}}$ the estimation error. Also, define the error $\boldsymbol{s} = \dot{\tilde{\boldsymbol{x}}} + \boldsymbol{P}\tilde{\boldsymbol{x}}$, where $\boldsymbol{P}$ is a positive definite diagonal matrix and $\tilde{\boldsymbol{x}} = \boldsymbol{x}_d - \boldsymbol{x}$. Then, for selected constant diagonal positive definite matrix $\boldsymbol{K}$ and constant positive definite matrix $\boldsymbol{\Gamma}_f$, the control law

$$\begin{aligned}\boldsymbol{\tau} = \boldsymbol{J}^T\boldsymbol{\Lambda}(\ddot{\boldsymbol{x}}_d + \boldsymbol{P}\dot{\tilde{\boldsymbol{x}}} - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) + \boldsymbol{J}^T\left(\frac{1}{2}\dot{\boldsymbol{\Lambda}} + \boldsymbol{K}\right) - \boldsymbol{J}^T\boldsymbol{J}^{\#T}\hat{\boldsymbol{\tau}} \\ + \boldsymbol{M}\boldsymbol{N}_{\#}(\ddot{\boldsymbol{q}}_d + \boldsymbol{M}^{-1}(\boldsymbol{B}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_d\tilde{\boldsymbol{q}})) + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}),\end{aligned} \tag{5}$$

with the disturbance observer

$$\dot{\hat{\boldsymbol{\tau}}} = -\boldsymbol{\Gamma}_f^T\boldsymbol{J}^{\#}\boldsymbol{s}, \tag{6}$$

guarantee that the $\tilde{\boldsymbol{x}}$ and $\dot{\tilde{\boldsymbol{x}}}$ go to zero asymptotically while the null-space impedance behavior is imposed. Moreover, the estimated disturbance remains bounded and the closed-loop system is stable □

In (5), $\boldsymbol{\Lambda} = (\boldsymbol{J}\boldsymbol{M}^{-1}\boldsymbol{J}^T)^{-1}$ is task inertia matrix, $\boldsymbol{J}^{\#} = \boldsymbol{M}^{-1}\boldsymbol{J}^T\boldsymbol{\Lambda}$ is the dynamically consistent generalized inverse (Khatib et al., 2004) and $\boldsymbol{N}_{\#} = (\boldsymbol{I} - \boldsymbol{J}^{\#}\boldsymbol{J})$.

Applying the above control law to the system (1) the closed-loop behavior of the robot in the task space is given by

$$\begin{aligned}\boldsymbol{J}^{\#T}\tilde{\boldsymbol{\tau}} = \tilde{\boldsymbol{f}} = \Lambda\dot{\boldsymbol{s}} + (\frac{1}{2}\dot{\boldsymbol{\Lambda}} + \boldsymbol{K})\boldsymbol{s}, \\ \tilde{\boldsymbol{f}} = \boldsymbol{f}_{ext} - \hat{\boldsymbol{f}},\end{aligned} \tag{7}$$

where $\hat{\boldsymbol{f}} = \boldsymbol{J}^{\#T}\hat{\boldsymbol{\tau}}$ is the estimation of $\boldsymbol{f}_{ext} = \boldsymbol{J}^{\#T}\boldsymbol{\tau}_{ext}$. The closed-loop dynamics for the null-space is

$$\boldsymbol{N}_{\#}(\ddot{\tilde{\boldsymbol{q}}} + \boldsymbol{M}^{-1}(\boldsymbol{B}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_d\tilde{\boldsymbol{q}}) - \boldsymbol{M}^{-1}\boldsymbol{\tau}_{ext}) = \boldsymbol{0}, \tag{8}$$

corresponding to an impedance behavior in the null-space of the main task.

The proof is based on the following Lyapunov-like function:

$$V = \tilde{\boldsymbol{x}}^T\boldsymbol{P}^T\boldsymbol{K}\tilde{\boldsymbol{x}} + \frac{1}{2}\boldsymbol{s}^T\boldsymbol{\Lambda}\boldsymbol{s} + \frac{1}{2}\tilde{\boldsymbol{\tau}}^T\boldsymbol{\Gamma}_f\tilde{\boldsymbol{\tau}}. \tag{9}$$

Notice that, while the force estimation error $\tilde{\boldsymbol{f}} = \boldsymbol{J}^{\#T}\tilde{\boldsymbol{\tau}}$ goes to zero, the torque estimation error $\tilde{\boldsymbol{\tau}}$ converges to zero only for non-redundant robots.

4 Momentum-Based Observer

Another method to ensure the correct execution of the main task during interaction is based on the collision detection algorithm presented by De Luca and Ferrajoli (2008). The basic concept is the computation of the n-dimensional residual vector

$$\boldsymbol{r}(t) = \boldsymbol{K}_I\left(\boldsymbol{p}(t) - \int_0^t (\boldsymbol{\tau} + \boldsymbol{C}^T(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{r}(\sigma))d\sigma\right), \tag{10}$$

where $\boldsymbol{p}(t) = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}$ is the robot generalized momentum and $\boldsymbol{K}_I$ is a positive definite diagonal matrix. This vector can be computed using the measured signals $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$, and the commanded torque $\boldsymbol{\tau}$, with initial conditions $\boldsymbol{r}(0) = \boldsymbol{0}$ and $\boldsymbol{p}(0) = \boldsymbol{0}$. The dynamics of $\boldsymbol{r}$ is

$$\dot{\boldsymbol{r}} = -\boldsymbol{K}_I\boldsymbol{r} - \boldsymbol{K}_I\boldsymbol{\tau}_{ext}, \tag{11}$$

corresponding to a filtered version of the real external torques, i.e. $\boldsymbol{r}(t) = \boldsymbol{\tau}_{ext}$. In the absence of interaction, assuming no noise and unmodeled disturbances, $\boldsymbol{r}(t) = \boldsymbol{0}$. As soon as interaction occurs, the components of $\boldsymbol{r}$ will raise exponentially and will reach to the value of $-\boldsymbol{\tau}_{ext}$.

A control algorithm similar to the one that was given by (5), using $\boldsymbol{r}$ in place of $\hat{\boldsymbol{\tau}}$, is adopted

$$\begin{aligned}\boldsymbol{\tau} = {} & \boldsymbol{J}^T\boldsymbol{\Lambda}(\ddot{\boldsymbol{x}}_d + \boldsymbol{K}_D\dot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_p\tilde{\boldsymbol{x}} - \dot{\boldsymbol{J}}\dot{\boldsymbol{q}}) - \boldsymbol{J}^T\boldsymbol{J}^{\#T}\boldsymbol{r} \\ & + \boldsymbol{M}\boldsymbol{N}_{\#}[\ddot{\boldsymbol{q}}_d + \boldsymbol{M}_d^{-1}(\boldsymbol{B}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_d\tilde{\boldsymbol{q}})] + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}),\end{aligned} \tag{12}$$

By defining the estimation error $\tilde{\boldsymbol{r}} = \boldsymbol{r} + \boldsymbol{\tau}_{ext}$, the closed-loop task dynamics is

$$\begin{aligned}\dddot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_D\ddot{\tilde{\boldsymbol{x}}} + \boldsymbol{K}_p\dot{\tilde{\boldsymbol{x}}} &= \boldsymbol{J}\boldsymbol{M}^{-1}\tilde{\boldsymbol{r}} \\ \dot{\tilde{\boldsymbol{r}}} &= -\boldsymbol{K}_I\tilde{\boldsymbol{r}} + \dot{\boldsymbol{\tau}}_{ext}.\end{aligned} \tag{13}$$

while the closed-loop dynamics for the null-space is the same as in (8). From the stability properties of the cascade systems (Panteley and Loria, 1998) it can be shown that when $\|\dot{\boldsymbol{\tau}}_{ext}\|$ is bounded, $\tilde{\boldsymbol{x}}$ is also bounded and specifically when $\dot{\boldsymbol{\tau}}_{ext} = 0$ the system is asymptotically stable and $\tilde{\boldsymbol{x}}, \dot{\tilde{\boldsymbol{x}}} \to \boldsymbol{0}$.

5 Numerical Simulation

The proposed approaches are verified in simulation on a 7 DOF KUKA lightweight arm, neglecting the presence of joint elasticity and dissipative

effects. The position of the end-effector is assumed as the main task. Therefore, the robot has 4 degrees of redundancy. A second-order inverse kinematics algorithm is used to compute the joint space desired trajectory for the null-space impedance, assuming a desired trajectory also for the end-effector orientation.

The main end-effector task is set as a straight line of about 1 m length. It is assumed that the manipulator hits an obstacle during its maneuver. The obstacle is moving with a constant speed of 2 cm/s. The interaction is modeled as an elastic contact with stiffness 1000 Nm/rad and happens at the fourth joint of the robot arm. A snapshot of the system during contact is reported on Fig 2.

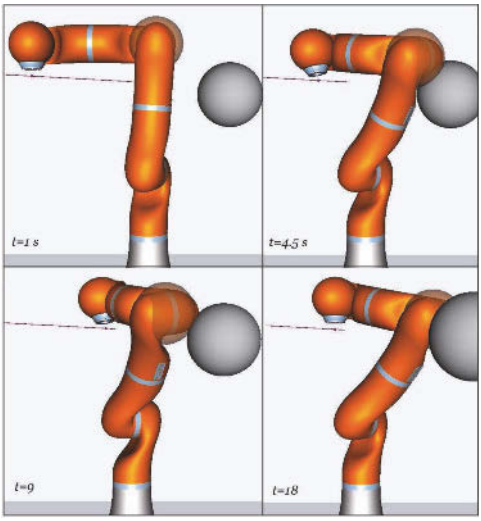


Figure 2. Snapshots of the system during interaction with an elastic sphere.

The position errors for the main task are reported in Fig. 3. It can be observed that the position errors are significantly reduced (of about one order of magnitude) when the observers are used. Notice that, during interaction, the end-effector orientation experiences errors since the orientation trajectory is assigned in the lower priority level as the desired trajectory for the null-space impedance. Thus the manipulator arm does not escape from the collision area and the redundancy allows to comply by keeping the position trajectory. As soon as the arm looses the contact with the obstacle, the orientation error disappears and the arm comes back to the desired configuration.

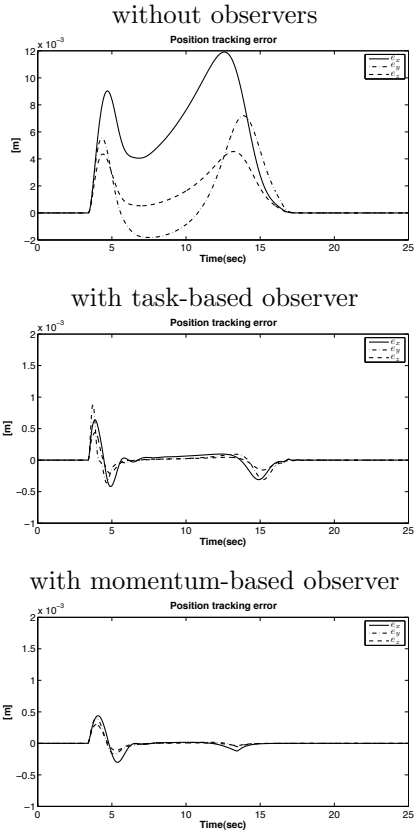


Figure 3. Position errors for the main task.

6 Conclusion

Two nonlinear control algorithms that ensure task space tracking error convergence, besides proper impedance behavior in the null-space, have been presented. The controllers do not need torque sensors and can be used for the case where the robot works close to humans and physical interaction occurs. The redundancy of the system is exploited to ensure safe and dependable physical interaction. The simulation results confirm the effectiveness of the proposed approach.[1]

[1]This work was partially funded by the European Community, within the FP7 ICT-287513 SAPHARI project and by the Italian Ministry of University and Research

Bibliography

A. Bicchi and G. Tonietti. Fast and soft arm tactics: dealing with the safety performance trade-off in robot arms design and control *IEEE Robotics and Automation Magazine*, vol. 11, pp. 22–33, 2004.

A. De Luca and L. Ferrajoli. Exploiting robot redundancy in collision detection and reaction. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3200–3305, 2008.

A. De Santis, B. Siciliano, A. De Luca and A. Bicchi. An atlas of physical human-robot interaction. *Mechanism and Machine Theory*, vol. 43, pp. 253–270, 2008.

S. Haddadin, A. Albu-Schäffer, A. De Luca, and G. Hirzinger. Collision detection and reaction: a contribution to safe physical human-robot interaction. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3356–3363, 2008.

O. Khatib, L. Sentis, J.H. Park and J. Warren. Whole-body dynamic behavior and control of human-like robots. *International Journal of Humanoid Robotics*, vol. 1, pp. 29–43, 2004.

V.J. Lumelsky and E. Cheung. Sensitive skin. *IEEE Sensors Journal*, vol. 1, pp. 41–51, 2001.

J. Nakanishi, R. Cory, M.J. Peters and S. Schaal. Operational space control: A theoretical and empirical comparison. *International Journal of Robotics Research*, vol. 27, pp. 737–757, 2008.

C. Ott. *Cartesian Impedance Control of Redundant and Flexible-Joint Robots.* Springer Tracts in Advanced Robotics, 2008.

E. Panteley and A. Loria. On global uniform asymptotic stability of nonlinear time-varying systems in cascade. *Systems and Control Letters*, vol. 33, pp. 131–138, 1998.

H. Sadeghian, L. Villani, M. Keshmiri and B. Siciliano. Multi-priority control in redundant robotic systems. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3752–3757, 2011.

B. Siciliano and J.J. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. *5th International Conference on Advanced Robotics*, pp. 1211–1216, 1991.

T. Yoshikawa and O. Khatib. Compliant motion control for a humanoid robot in contact with the environment and humans. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 211–218, 2008

within the PRIN 2009 20094WTJ29_003 ROCOCO' project.

Wrench Recovery for Wire-Actuated Parallel Manipulators

Leila Notash
Queen's University, Kingston, ON, Canada

Abstract Methodologies for investigating the effect of wire/actuator failures on the wrench capability of wire-actuated parallel manipulators are presented that result in minimum norm solution for the correctional wire tension and overall tension vectors. A planar manipulator is simulated to illustrate the proposed methodologies.

1 Introduction

In wire/cable-actuated manipulators, the motion of mobile platform (end effector) is controlled by the wires/cables that connect the platform to the base. Because wires act in tension and cannot exert forces in both directions along their lines of action, to fully constrain an m degrees of freedom (DOF) rigid body suspended by wires, in the absence of gravity and external force/moment (wrench), the number of wires/actuators of the manipulator should be larger than the DOF of manipulator (Figures 1). In this article, when the number of wires n is equal to $m + 1$, the manipulator will be referred to as "non-redundant". For redundant actuation, one or more additional wires could be included.

Wire-actuated parallel manipulators could fail because of the failure of a wire, sensor, actuator, or transmission mechanism. These failures could result in the loss of DOF, actuation, motion constraint, and information (Notash and Huang, 2003). From the force point of view, the failure of manipulator occurs if the wire does not provide the required force/torque, e.g., when the actuator force/torque is lost partially or fully or the actuator is saturated. This could also happen when the wire is broken or slack (zero tension), wire is jammed (constant length), or its actuating mechanism malfunctions such that a different wire force is provided.

Failure recovery from the actuator torque loss in closed-loop manipulators was investigated in (Ting et al., 1995). In (Notash, 2000), the effect of

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_25,

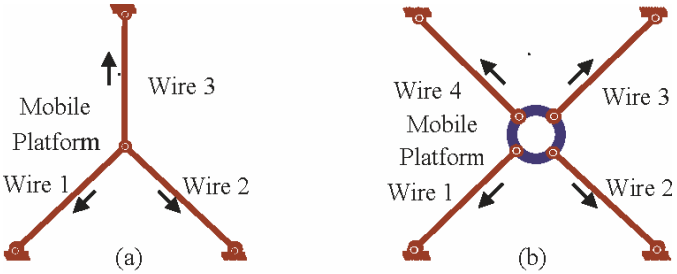


Figure 1. Planar wire-actuated manipulators (a) 2 DOF; and (b) 3 DOF.

redundancy in joint displacement sensing was investigated to facilitate the joint sensor fault detection, isolation and recovery. In (Chen et al., 2003), the task space was partitioned to complete the major task and optimize a secondary goal such as actuator fault tolerance. In (Roberts et al., 1998), the effect of losing a wire on the null space of the Jacobian matrix of a planar manipulator was considered.

In this article, failure of wire-actuated parallel manipulators is studied. Methodologies for recovering the lost wrench capability due to the failure of wires and actuators are presented in Section 2. Simulation results are reported in Section 3. The article concludes with Section 4.

2 Wrench Recovery Methodologies

For the n-wire-actuated parallel manipulators, the $n \times 1$ vector of wire forces $\boldsymbol{\tau} = \begin{bmatrix} \tau_1 & \cdots & \tau_n \end{bmatrix}^T$ is related to the $m \times 1$ vector of forces and moments (wrench) $\mathbf{F}$ applied by the platform with the $m \times n$ transposed Jacobian matrix $\mathbf{J}^T$ as

$$\mathbf{F} = \mathbf{J}^T \boldsymbol{\tau} = \begin{bmatrix} \mathbf{J}_1^T & \mathbf{J}_2^T & \cdots & \mathbf{J}_i^T & \cdots & \mathbf{J}_{n-1}^T & \mathbf{J}_n^T \end{bmatrix} \boldsymbol{\tau} = \sum_{j=1}^{n} \mathbf{J}_j^T \tau_j \tag{1}$$

where $m \leq 6$ depending on the dimension of task space. Column j of $\mathbf{J}^T$, $\mathbf{J}_j^T$, corresponds to the wrench applied on the platform by the j-th wire/actuator. The solution of $\mathbf{F} = \mathbf{J}^T \boldsymbol{\tau}$ for the wire tensions is

$$\boldsymbol{\tau} = \mathbf{J}^{\#T}\mathbf{F} + \left(\mathbf{I} - \mathbf{J}^{\#T}\mathbf{J}^T\right) \mathbf{k} = \mathbf{J}^{\#T}\mathbf{F} + \mathbf{N}\,\boldsymbol{\lambda} \tag{2}$$

where $\mathbf{J}^{\#T}$ is the generalized (Moore-Penrose) inverse of $\mathbf{J}^T$, $\mathbf{J}^{\#T}\mathbf{F}$ is the minimum norm (particular) solution, and $\left(\mathbf{I} - \mathbf{J}^{\#T}\mathbf{J}^T\right)\mathbf{k}$ and $\mathbf{N}\,\boldsymbol{\lambda}$ represent the homogeneous solution. $\left(\mathbf{I} - \mathbf{J}^{\#T}\mathbf{J}^T\right)\mathbf{k}$ is the projection of the arbitrary vector $\mathbf{k}$ onto the null space of $\mathbf{J}^T$, and the columns of $n \times (n-m)$ matrix $\mathbf{N}$ correspond to the orthonormal basis of the null space of $\mathbf{J}^T$. The minimum

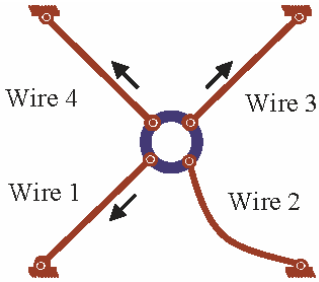


Figure 2. Four-wire-actuated parallel manipulator with slack wire 2.

norm solution could result in negative value for a wire tension, then the $n \times 1$ vector $\mathbf{k}$, (or the $(n-m) \times 1$ vector $\boldsymbol{\lambda}$) is chosen such that positive tension is maintained in the wires. Detailed discussions on the conditions for ensuring positive tension in wires are presented in (Notash, 2011).

2.1 Methodology 1 – Range Space of $\mathbf{J}^{\#T}$ after Failure

When wire i is failed, its tension τ_{ci} will be different than the desired value τ_i. When wire i is broken or slack (Figure 2) $\tau_{ci} = 0$. Wire i could also have a non-zero input, $\tau_{ci} \neq 0$, different than τ_i, e.g., when the wire tension or the actuator torque reaches the limit (maximum value). Then, the platform wrench will be

$$\mathbf{F}_f = \begin{bmatrix}\mathbf{J}_1^T & \mathbf{J}_2^T & \cdots & \mathbf{J}_i^T & \cdots & \mathbf{J}_{n-1}^T & \mathbf{J}_n^T\end{bmatrix} \boldsymbol{\tau}_f \quad = \sum_{j=1}^{n} \mathbf{J}_j^T \tau_j - \mathbf{J}_i^T(\tau_i - \tau_{ci}) \tag{3}$$

where $\boldsymbol{\tau}_f = [\tau_1 \ \tau_2 \ \ldots \ \tau_{ci} \ \ldots \ \tau_{n-1} \ \tau_n]^T$ and $\tau_i - \tau_{ci}$ is the lost wire force due to failure of wire i. When $\tau_{ci} \neq \tau_i$, to fully recover the lost wrench and maintain $\mathbf{F}$, the remaining (healthy) wires must provide the required wrench. With the correctional force provided by the remaining wires $\boldsymbol{\tau}_{corr} = [\tau_{corr1} \ \tau_{corr2} \ \ldots \ 0 \ \ldots \ \tau_{corrn-1} \ \tau_{corrn}]^T$, the recovered wrench will be

$$\mathbf{F}_r = \mathbf{J}^T \boldsymbol{\tau}_f + \mathbf{J}_f^T \boldsymbol{\tau}_{corr} \tag{4}$$

where entry i of $\boldsymbol{\tau}_{corr}$ and column i of the Jacobian matrix after failure $\mathbf{J}_f^T = \begin{bmatrix}\mathbf{J}_1^T & \mathbf{J}_2^T & \cdots & \mathbf{0} & \cdots & \mathbf{J}_{n-1}^T & \mathbf{J}_n^T\end{bmatrix}$ are replaced by zeros. Then, the change in the platform wrench will be $\mathbf{F} - \mathbf{F}_r = \mathbf{J}^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f) - \mathbf{J}^T\boldsymbol{\tau}_{corr}$. When the forces/torques of k wires/actuators are different than the corresponding desired values the lost wrench is $\sum \mathbf{J}_i^T(\tau_i - \tau_{ci}) = \mathbf{J}^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f)$, where the summation is taken over the failed wires/actuators.

To fully compensate for the lost wrench, i.e., for $\mathbf{F} - \mathbf{F}_r = \mathbf{0}$, the correctional force provided by the remaining wires should be (Notash, 2011)

$$\boldsymbol{\tau}_{corr} = \mathbf{J}_f^{\#T} \sum \mathbf{J}_i^T (\tau_i - \tau_{ci}) = \mathbf{J}_f^{\#T} \mathbf{J}^T (\boldsymbol{\tau} - \boldsymbol{\tau}_f) \tag{5}$$

where in $\mathbf{J}_f^T$ k columns of $\mathbf{J}^T$, corresponding to the wires/actuators with different inputs, are replaced by zeros. Then, the overall wire force will be

$$\boldsymbol{\tau}_{tot} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr} = \mathbf{J}_f^{\#T} \mathbf{J}^T \boldsymbol{\tau} + (\mathbf{I} - \mathbf{J}_f^{\#T} \mathbf{J}^T) \boldsymbol{\tau}_f = \mathbf{J}_f^{\#T} \mathbf{J}^T \boldsymbol{\tau} \tag{6}$$

If $\mathbf{J}_f^T$ has full row-rank, i.e., $\mathbf{F}$ belongs to the range space of $\mathbf{J}_f^T$, $\mathbf{F} \in \Re(\mathbf{J}_f^T)$, the generalized inverse (GI) of $\mathbf{J}_f^T$ is $\mathbf{J}_f^{\#T} = \mathbf{J}_f \left(\mathbf{J}_f^T \mathbf{J}_f \right)^{-1}$ as the vector of wire forces is physically consistent (entries have dimension of force). When the rank of $\mathbf{J}_f^T$ is less than m, in general, the lost wrench cannot be fully recovered. In this case, if $\mathbf{J}_f^T$ has full column-rank and the lost wrench is not in the range space of $\mathbf{J}_f^T$, the platform wrench that best approximates the lost wrench in the least-square sense is calculated using the weighted left-GI of $\mathbf{J}_f^T$, $\mathbf{J}_f^{\#^T} = \left(\mathbf{J}_f \mathbf{W}_F \mathbf{J}_f^T \right)^{-1} \mathbf{J}_f \mathbf{W}_F$. The weighting metric $\mathbf{W}_F$ is chosen such that $\mathbf{F}^T (\mathbf{W}_F \mathbf{F})$ becomes physically consistent.

With the correctional input from healthy wires, when $\mathbf{F} \in \Re(\mathbf{J}_f^T)$ the condition for partial recovery of the lost wrench after failure is

$$\mathbf{F}_{\Re^\perp} = \left(\mathbf{I} - \mathbf{J}_f^T \mathbf{J}_f^{\#T} \right) \mathbf{F} \neq \mathbf{0} \tag{7}$$

That is, the lost wrench will be fully recovered when all the components of $\mathbf{F}$ projected onto the orthogonal complement of the range space of $\mathbf{J}_f^T$ are zero and the conditions for positive tension in wires are met.

2.2 Minimum Norm Input Vectors

When the number of failed wires/actuators is equal to or less than the number of redundant wires/actuators, i.e., $k \leq n - m - 1$, provided the manipulator is not at a singularity, after removing the columns of the $m \times n$ matrix $\mathbf{J}^T$ corresponding to failed wires/actuators, the reduced $m \times (n-k)$ Jacobian matrix with rank m will result in infinite solutions for $\boldsymbol{\tau}_{corr}$ and $\boldsymbol{\tau}_{tot} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr}$ as long as the pose is in the wrench closure workspace of manipulator. In the following subsections, expressions for $\boldsymbol{\tau}_{corr}$ and $\boldsymbol{\tau}_{tot}$ will be derived considering their Euclidian norms.

Methodology 2 – Minimum Norm for Correctional Tension Vector. For optimum tension distribution and to minimize the jump in the tension after failure of k out of n wires/actuators, the objective function is the Euclidian norm of the correctional tension vector provided by the

remaining wires/actuators, $\boldsymbol{\tau}_{corr} \cdot \boldsymbol{\tau}_{corr} = \tau_{corr1}^2 + \tau_{corr2}^2 + \cdots + \tau_{corrn}^2 = \sum_{n-k} \tau_{corrj}^2$. The linear constraint equation in terms of the overall wire tension vector is $\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_{tot} = \mathbf{0}$. The Lagrange function L is formulated by augmenting the constraint equation with the objective function using the Lagrange multiplier vector $\boldsymbol{\lambda}$

$$L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda}) = \frac{1}{2} \boldsymbol{\tau}_{corr} \cdot \boldsymbol{\tau}_{corr} - \boldsymbol{\lambda}^T (\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_{tot}) \tag{8}$$

If $\boldsymbol{\tau}_{corr} \cdot \boldsymbol{\tau}_{corr}$ is a minimum for the original constrained problem, then at the stationary point, the gradient of L vanishes, i.e., $\nabla L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda}) = \mathbf{0}$.

When $\tau_{ci} = 0$ the constraint equation is $\mathbf{F} - \mathbf{J}_f^T \boldsymbol{\tau}_{tot} = \mathbf{0}$ and

$$\frac{\partial L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\tau}_{corr}} = \boldsymbol{\tau}_{corr} - \mathbf{J}_f \boldsymbol{\lambda} = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\tau}_{corr} = \mathbf{J}_f \boldsymbol{\lambda} \tag{9}$$

$$\frac{\partial L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \mathbf{J}_f^T \boldsymbol{\tau}_{tot} - \mathbf{F} = \mathbf{0} \quad \Rightarrow \quad \mathbf{J}_f^T (\boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr}) = \mathbf{F} \tag{10}$$

Then, $\boldsymbol{\lambda} = (\mathbf{J}_f^T \mathbf{J}_f)^{-1} (\mathbf{F} - \mathbf{J}_f^T \boldsymbol{\tau}_f)$ and because $\mathbf{J}_f^T \boldsymbol{\tau}_f = \mathbf{J}^T \boldsymbol{\tau}_f$ when $\tau_{ci} = 0$

$$\boldsymbol{\tau}_{corr} = \mathbf{J}_f (\mathbf{J}_f^T \mathbf{J}_f)^{-1} (\mathbf{F} - \mathbf{J}_f^T \boldsymbol{\tau}_f) = \mathbf{J}_f^{\#T} \mathbf{J}^T (\boldsymbol{\tau} - \boldsymbol{\tau}_f) \tag{11}$$

When $\tau_{ci} \neq 0$ the constraint equation is $\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_{tot} = \mathbf{0}$. To ensure zero correctional tension for the failed wires, $\mathbf{J}\boldsymbol{\lambda}$ is replaced with $\mathbf{J}_f \boldsymbol{\lambda}$ for $\boldsymbol{\tau}_{corr}$

$$\frac{\partial L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\tau}_{corr}} = \boldsymbol{\tau}_{corr} - \mathbf{J} \boldsymbol{\lambda} = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\tau}_{corr} = \mathbf{J}_f \boldsymbol{\lambda} \tag{12}$$

$$\frac{\partial L(\boldsymbol{\tau}_{corr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \mathbf{J}^T \boldsymbol{\tau}_{tot} - \mathbf{F} = \mathbf{0} \quad \Rightarrow \quad \mathbf{J}^T (\boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr}) = \mathbf{F} \tag{13}$$

Then, $\boldsymbol{\lambda} = (\mathbf{J}^T \mathbf{J}_f)^{-1} (\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_f)$ and because $\mathbf{J}_f (\mathbf{J}^T \mathbf{J}_f)^{-1} = \mathbf{J}_f (\mathbf{J}_f^T \mathbf{J}_f)^{-1}$

$$\boldsymbol{\tau}_{corr} = \mathbf{J}_f (\mathbf{J}^T \mathbf{J}_f)^{-1} (\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_f) = \mathbf{J}_f^{\#T} \mathbf{J}^T (\boldsymbol{\tau} - \boldsymbol{\tau}_f) \tag{14}$$

As indicated by equations (11) and (14), the first failure recovery methodology results in the minimum 2-norm solution for $\boldsymbol{\tau}_{corr}$.

Methodology 3 – Minimum Norm for Overall Tension Vector. For optimum tension distribution and to minimize the actuation energy after failure, the objective function is the square of the Euclidian norm of the overall tension vector $\boldsymbol{\tau}_{tot}$, and the linear constraint equation in terms of $\boldsymbol{\tau}_{tot}$ is $\mathbf{F} - \mathbf{J}^T \boldsymbol{\tau}_{tot} = \mathbf{0}$. The Lagrange function L is

$$L(\boldsymbol{\tau}_{tot}, \boldsymbol{\lambda}) = \frac{1}{2}\boldsymbol{\tau}_{tot} \cdot \boldsymbol{\tau}_{tot} - \boldsymbol{\lambda}^T(\mathbf{F} - \mathbf{J}^T\boldsymbol{\tau}_{tot}) \tag{15}$$

and when $\boldsymbol{\tau}_{tot} \cdot \boldsymbol{\tau}_{tot}$ is a minimum for the original problem $\nabla L(\boldsymbol{\tau}_{tot}, \boldsymbol{\lambda}) = \mathbf{0}$.

When $\tau_{ci} = 0$, to have zero overall (and correctional) tension for the failed (slack) wires, the constraint equation is $\mathbf{F} - \mathbf{J}_r^T\boldsymbol{\tau}_{totr} = \mathbf{0}$, where the reduced Jacobian matrix $\mathbf{J}_r^T$ and reduced overall wire tension vector $\boldsymbol{\tau}_{totr}$ are respectively obtained by removing the columns and entries of $\mathbf{J}^T$ and $\boldsymbol{\tau}_{tot}$ corresponding to the failed wires. Consequently

$$\frac{\partial L(\boldsymbol{\tau}_{totr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\tau}_{totr}} = \boldsymbol{\tau}_{totr} - \mathbf{J}_r\boldsymbol{\lambda} = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\tau}_{totr} = \mathbf{J}_r\boldsymbol{\lambda} \tag{16}$$

$$\frac{\partial L(\boldsymbol{\tau}_{totr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \mathbf{J}_r^T\boldsymbol{\tau}_{totr} - \mathbf{F} = \mathbf{0} \quad \Rightarrow \quad \mathbf{J}_r^T\boldsymbol{\tau}_{totr} = \mathbf{F} \tag{17}$$

Then, $\boldsymbol{\lambda} = (\mathbf{J}_r^T\mathbf{J}_r)^{-1}\mathbf{F}$ and

$$\boldsymbol{\tau}_{totr} = \mathbf{J}_r^{\#T}\mathbf{F} = \mathbf{J}_r^{\#T}\mathbf{J}^T\boldsymbol{\tau} \tag{18}$$

The correctional tension from the healthy wires will be $\boldsymbol{\tau}_{corrr} = \boldsymbol{\tau}_{totr} - \boldsymbol{\tau}_{fr}$, where the reduced wire tension vector after failure $\boldsymbol{\tau}_{fr}$ is obtained by removing the zero entries of $\boldsymbol{\tau}_f$ corresponding to the failed wires.

When $\tau_{ci} \neq 0$, to calculate the minimum norm $\boldsymbol{\tau}_{tot}$, first the portion of the platform wrench provided by the failed wires with $\tau_{ci} \neq 0$ should be removed from the required wrench. When kc out of k failed wires have non-zero tension $\mathbf{F}^* = \mathbf{F} - \sum_{kc}\mathbf{J}_l^T\boldsymbol{\tau}_{cl}$ and the Lagrange function and its partial derivatives are

$$L(\boldsymbol{\tau}_{totr}, \boldsymbol{\lambda}) = \frac{1}{2}\boldsymbol{\tau}_{totr} \cdot \boldsymbol{\tau}_{totr} - \boldsymbol{\lambda}^T(\mathbf{F}^* - \mathbf{J}_r^T\boldsymbol{\tau}_{totr}) \tag{19}$$

$$\frac{\partial L(\boldsymbol{\tau}_{totr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\tau}_{totr}} = \boldsymbol{\tau}_{totr} - \mathbf{J}_r\boldsymbol{\lambda} = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\tau}_{totr} = \mathbf{J}_r\boldsymbol{\lambda} \tag{20}$$

$$\frac{\partial L(\boldsymbol{\tau}_{totr}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \mathbf{J}_r^T\boldsymbol{\tau}_{totr} - \mathbf{F}^* = \mathbf{0} \quad \Rightarrow \quad \mathbf{J}_r^T(\boldsymbol{\tau}_{totr}) = \mathbf{F}^* \tag{21}$$

Then, $\boldsymbol{\lambda} = (\mathbf{J}_r^T\mathbf{J}_r)^{-1}\mathbf{F}^*$ and the overall tension of healthy wires is

$$\boldsymbol{\tau}_{totr} = \mathbf{J}_r^{\#T}\mathbf{F}^* = \mathbf{J}_r^{\#T}\left(\mathbf{J}^T\boldsymbol{\tau} - \sum_{kc}\mathbf{J}_l^T\boldsymbol{\tau}_{cl}\right) \tag{22}$$

$\boldsymbol{\tau}_{tot}$ is obtained by incorporating the tension of the failed wires in $\boldsymbol{\tau}_{totr}$. In view of equations (18) and (22), the failure recovery methodology of Section 2.1 results in the minimum 2-norm solution for $\boldsymbol{\tau}_{tot}$ as well.

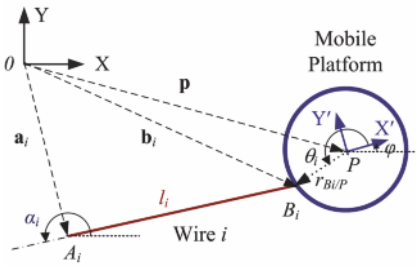


Figure 3. Parameters of planar wire-actuated parallel manipulators.

3 Case Study

The mobile platform of planar wire-actuated parallel manipulators is connected to the base by n wires, each wire with a length of l_i and orientation of α_i (Figure 3). The attachment points of wire i to the base and platform are denoted as points A_i and B_i, respectively. The angular positions of points B_i on the platform are denoted by θ_i.

For the 2 DOF translational manipulators, the wire forces $\boldsymbol{\tau}$ is related to the platform wrench $\mathbf{F}$ by the Jacobian matrix as

$$\mathbf{F} = \mathbf{J}^T\boldsymbol{\tau} = \begin{bmatrix} \cos\alpha_1 & \cdots & \cos\alpha_n \\ \sin\alpha_1 & \cdots & \sin\alpha_n \end{bmatrix} \begin{bmatrix} \tau_1 \\ \vdots \\ \tau_n \end{bmatrix} \tag{23}$$

For $n = 4$, coordinates of A_i, $i = 1, \ldots, 4$, as (–2, –1.5), (2, –1.5), (2, 1.5) and (–2, 1.5), respectively, and platform pose of $\mathbf{p} = [1 \quad 0]^T$ meter

$$\mathbf{J}^T = \begin{bmatrix} \mathbf{J}_1^T & \mathbf{J}_2^T & \mathbf{J}_3^T & \mathbf{J}_4^T \end{bmatrix} = \begin{bmatrix} -0.894 & 0.555 & 0.555 & -0.894 \\ -0.447 & -0.832 & 0.832 & 0.447 \end{bmatrix} \tag{24}$$

A null space vector of $\mathbf{J}^T$ is $\mathbf{N} = [0.874 \quad 0.591 \quad 1.000 \quad 0.113]^T$, with positive entries. Therefore, this pose is in the wrench closure workspace of manipulator. For $\mathbf{F} = [-16.986 \quad 21.737]^T$ Newtons, the minimum norm vector of wire forces is $\boldsymbol{\tau} = \mathbf{J}^{\#T}\mathbf{F} = [1.411 \quad -14.388 \quad 5.881 \quad 12.305]^T$ with a magnitude of $\|\boldsymbol{\tau}\|_2 = 19.875$ and a negative tension for wire 2. The tension of wire 2 could be adjusted to a threshold positive value by using the homogeneous solution or employing the methodologies of Section 2. Utilizing the first methodology and treating wire 2 as failed, when $\tau_{c2} = 2$ then $\mathbf{F}_f = \mathbf{J}^T\boldsymbol{\tau}_f = [-7.896 \quad 8.102]^T$.

The wire forces are adjusted for full recovery. Hence, the minimum norm correctional force to be provided by wires 1, 3 and 4 is

$$\boldsymbol{\tau}_{corr} = \mathbf{J}_f^{\#T}\mathbf{J}_2^T(\tau_2 - \tau_{c2}) = [0.535 \quad 0 \quad 8.625 \quad 14.977]^T \tag{25}$$

with $\|\boldsymbol{\tau}_{corr}\|_2 = 17.292$. The minimum norm overall wire forces will be

$$\boldsymbol{\tau}_{tot} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr} = [1.946 \quad 2.000 \quad 14.507 \quad 27.282]^T \tag{26}$$

with $\|\boldsymbol{\tau}_{tot}\|_2 = 31.025$, which produces the original wrench. The minimum norm for $\boldsymbol{\tau}_{tot}$ gives the same results, i.e.,

$$\boldsymbol{\tau}_{totr} = \mathbf{J}_r^{\#T}\mathbf{F}^* = [1.946 \quad 14.507 \quad 27.282]^T \tag{27}$$

If at this pose, wire 4 is failed (slack), i.e., for $\tau_{c2} = 2$ and $\tau_{c4} = 0$, the correctional force will be $\boldsymbol{\tau}_{corr} = [55.000 \quad 0 \quad 52.617 \quad 0]^T$ with $\|\boldsymbol{\tau}_{corr}\|_2 = 76.187$, and $\boldsymbol{\tau}_{tot} = [56.511 \quad 2.000 \quad 58.498 \quad 0]^T$ with $\|\boldsymbol{\tau}_{tot}\|_2 = 81.360$.

4 Conclusion

Failure analysis of wire-actuated parallel manipulators was discussed in this paper and three methodologies were presented for recovering the lost wrench. The first methodology is based on the projection of the lost wrench due to wire failure onto the orthogonal complement of the null space of the Jacobian matrix of the failed manipulator. The second and third methodologies are based on the Lagrange multiplier method for minimizing the norms of the correctional and overall wire force vectors, respectively. These methodologies are also applicable for the cases that the minimum norm solution for the vector of wire tensions gives negative value for a wire tension, by treating the corresponding wire as failed. It was presented that all three methodologies produce the same result.

Bibliography

L. Notash, and L. Huang, On the Design of Fault Tolerant Parallel Manipulators, Mech. and Machine Theory, 38:85–101, 2003.

Y. Ting, S. Tosunoglu, and R. Freeman, Torque Redistribution and Time Regulation Methods for Actuator Saturation Avoidance of Fault-Tolerant Parallel Robots, J. Robotic Systems, 12:807–820, 1995.

L. Notash, Joint Sensor Fault Detection for Fault Tolerant Parallel Manipulators, J. Robotic Systems, 17:149–157, 2000.

Y. Chen, J.E. McInroy, and Y. Yi, Optimal, Fault-Tolerant Mappings to Achieve Secondary Goals Without Compromising Primary Performance, IEEE Trans. Rob. Automation, 19: 680–691, 2003.

R.G. Roberts, T. Graham, , and T. Lippitt, On the Inverse Kinematics, Statics, and Fault Tolerance of Cable-Suspended Robots, J. Robotic Systems, 15:581–597, 1998.

L. Notash, Failure Recovery for Wrench Capability of Wire-Actuated Parallel Manipulators, Robotica, doi:10.1017/S0263574711001160, 2011.

Wrench Recovery of Parallel Manipulators for Full Retrieval of Subtasks

Vahid Nazari * and Leila Notash *

* Department of Mechanical and Materials Engineering, Queens University, Kingston, ON, Canada
[nazariv, notash]@me.queensu.ca

Abstract In this paper, the wrench recovery of parallel manipulators after actuator failure is investigated. To achieve the desired wrench capability, the mobile platform task is divided into the recoverable and non-recoverable subtasks. The presented work is based on the projection of the lost wrench due to total or partial actuator failure onto the range space of the reduced Jacobian matrix. The force/torque of the healthy joints is adjusted such that the secondary goal comprising the 2-norm of the error of non-recoverable wrench and the 2-norm of the vector of the overall forces/torques of the healthy joints is minimized without affecting the recovered components of the wrench.

1 Introduction

Parallel manipulators consist of a mobile platform that is connected to a base platform through several legs/branches, each leg comprising links and active and passive joints. The advantages of parallel manipulators over the serial manipulators are large load capacity, low inertia, high acceleration, high accuracy, and high stiffness. However, small useful workspace is their drawbacks. Manipulators employed in hazardous or remote environments, where failure can lead to catastrophic consequences, are recommended to be fault tolerant. One approach in designing fault tolerant manipulators is utilizing redundancy. Redundancy in manipulators offers many advantages over non-redundant manipulators including optimizing fault tolerance measures Roberts and Maciejewski (1996), recovering lost force/moment (wrench) and motion Notash (2011a,b),obstacle avoidance Jamisola, Maciejewski and Roberts (2003, 2004) and singularity avoidance Yoshikawa (1984). The effect of actuator/joint failures, including zero and non-zero force/torque, on the wrench capabilities of parallel manipulators was investigated in Notash (2011a). Criteria were established to examine full and

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_26,

partial recovery of the lost wrench based on the projection of the lost joint force/torque onto the orthogonal complement of the null space of the reduced Jacobian matrix, which is the same as its range space. In Chen, McInroy and Yi (2003) the mobile platform task was partitioned into the major and secondary subtasks. The major subtasks were controlled as precisely as possible and the secondary subtasks were compromised to optimize a secondary goal.

In this paper, the wrench recovery of parallel manipulators is studied. Once the actuator failures are detected, the mobile platform task is divided into the recoverable and non-recoverable subtasks based on the number, type and location of the failed joint(s); the manipulator configuration and the task/application requirements. In Section 2, the failure recovery methodology for the lost wrench, due to failure in actuators, reported in Notash (2011a) is reviewed. In Section 3, the task assigned to the mobile platform is divided into the recoverable and non-recoverable subtasks based on the projection of the lost wrench onto the range space of the reduced Jacobian matrix. The static force modeling of a Stewart-Gough manipulator and simulation results are reported in Section 4. Finally conclusion is given in Section 5.

2 Failure Recovery

Considering the static force model in parallel manipulators, the wrench applied by the mobile platform, $\mathbf{F} = [F_x \; F_y \; F_z \; M_x \; M_y \; M_z]^T$, is related to the joint force/torque using the transposed Jacobian matrix as

$$\mathbf{F} = \mathbf{J}^T \boldsymbol{\tau} = \begin{bmatrix} \mathbf{J}_1^T & \mathbf{J}_2^T & \dots & \mathbf{J}_i^T & \dots & \mathbf{J}_{n-1}^T & \mathbf{J}_n^T \end{bmatrix} \boldsymbol{\tau} \tag{1}$$

where $\mathbf{F}$ is a 6×1 vector, $\boldsymbol{\tau}$ is an $n \times 1$ vector, $\mathbf{J}^T$ is a $6 \times n$ matrix and n is the number of active joints. Column j of $\mathbf{J}^T$ corresponds to the wrench applied on the mobile platform by active joint j, and row k of $\mathbf{J}^T$ represents the contribution of all active joints to generate entry k of the wrench.

As reported in Notash (2011a), when the active joint h fails, the manipulator may not be able to provide the required task. In general, the failed joint has different force/torque, τ_{ch}, than the required value, τ_h. The mobile platform wrench after the failure will be $\mathbf{F}_f = [\mathbf{J}_1^T \; \mathbf{J}_2^T \; \dots \; \mathbf{J}_i^T \; \dots \; \mathbf{J}_n^T]\boldsymbol{\tau}_f = \mathbf{J}^T\boldsymbol{\tau}_f$ where $\boldsymbol{\tau}_f = \begin{bmatrix} \tau_1 & \tau_2 & \dots & \tau_{ch} & \dots & \tau_n \end{bmatrix}^T$. The lost wrench due to failure in joint h will be $\Delta\mathbf{F}_f = \mathbf{F} - \mathbf{F}_f = \mathbf{J}^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f) = \mathbf{J}_h^T(\tau_h - \tau_{ch})$. If the actuator of the failed joint provides force/torque equal to the required input then the failure has no effect on the wrench capability of the manipulator. Otherwise, the force/torque of the healthy joints should be adjusted such that the lost wrench is compensated. The correctional force/torque is

added to the force/torque of the healthy joints to fully or partially recover the wrench of the mobile platform as Notash (2011a)

$$\mathbf{F}_r = \mathbf{J}^T \boldsymbol{\tau}_f + \mathbf{J}^T \Delta\boldsymbol{\tau}_{corr} = \mathbf{J}^T \boldsymbol{\tau}_f + \mathbf{J}_f^T \Delta\boldsymbol{\tau}_{corr} \tag{2}$$

where $\Delta\boldsymbol{\tau}_{corr} = [\Delta\tau_1 \ \Delta\tau_2 \ \ldots \ 0 \ \ldots \ \Delta\tau_{n-1} \ \Delta\tau_n]^T$ and the reduced Jacobian matrix of the manipulator is defined as $\mathbf{J}_f^T = [\mathbf{J}_1^T \ \mathbf{J}_2^T \ \ldots \ 0 \ \ldots \ \mathbf{J}_n^T]$ in which the column associated with the failed joint is replaced by a 6×1 zero vector. The change in the wrench of the mobile platform after applying the correctional force/torque is

$$\begin{aligned} \Delta\mathbf{F}_r &= \mathbf{F} - \mathbf{F}_r = \mathbf{J}_h^T(\boldsymbol{\tau}_h - \boldsymbol{\tau}_{ch}) - \mathbf{J}_f^T \Delta\boldsymbol{\tau}_{corr} \\ &= \mathbf{J}^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f) - \mathbf{J}_f^T \Delta\boldsymbol{\tau}_{corr} = \Delta\mathbf{J}^T \boldsymbol{\tau} - \mathbf{J}_f^T \Delta\boldsymbol{\tau}_{corr} \end{aligned} \tag{3}$$

where $\Delta\mathbf{J}^T = \mathbf{J}^T - \mathbf{J}_f^T = [\mathbf{0} \ \mathbf{0} \ \ldots \ \mathbf{J}_h^T \ \ldots \ \mathbf{0} \ \mathbf{0}] \in \mathbb{R}^{6\times n}$. To fully compensate for the lost mobile platform wrench, i.e., $\Delta\mathbf{F}_r = \mathbf{0}$, the correctional input provided by the healthy joints should be

$$\begin{aligned} \Delta\boldsymbol{\tau}_{corr} &= \mathbf{J}_f^{\#T}\mathbf{J}^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f) + (\mathbf{I} - \mathbf{J}_f^{\#T}\mathbf{J}_f^T)\mathbf{k}_1 \\ &= \mathbf{J}_f^{\#T}\Delta\mathbf{J}^T\boldsymbol{\tau} + (\mathbf{I} - \mathbf{J}_f^{\#T}\mathbf{J}_f^T)\mathbf{k}_1 \end{aligned} \tag{4}$$

The first term on the right hand side of equation (4) represents the minimum 2-norm solution of the vector of the correctional force/torque. The second term is the homogeneous solution that maps an arbitrary vector $\mathbf{k}_1 \in \mathbb{R}^{n\times 1}$ onto the null space of $\mathbf{J}_f^T$ after the failure. If the number of actuators is greater than the task space dimension and $\mathbf{J}_f^T$ has full row-rank and the manipulator has a combination of active revolute joints and active prismatic joints, then a weighting matrix $\mathbf{W}_\tau$ on $\boldsymbol{\tau}$ would be required to calculate the weighted generalized inverse of $\mathbf{J}_f^T$ as

$$\mathbf{J}_f^{\#T} = \mathbf{W}_\tau^{-1}\mathbf{J}_f(\mathbf{J}_f^T\mathbf{W}_\tau^{-1}\mathbf{J}_f)^{-1} \tag{5}$$

The positive-definite matrix $\mathbf{W}_\tau$ is selected such that $\boldsymbol{\tau}^T\mathbf{W}_\tau\boldsymbol{\tau}$ becomes physically consistent. If the vector of the joint forces/torques is physically consistent the generalized (Moore-Penrose) inverse $\mathbf{J}_f^{+T} = \mathbf{J}_f(\mathbf{J}_f^T\mathbf{J}_f)^{-1}$ has physical meaning Notash (2011a). When $\mathbf{J}_f^T$ has neither full row-rank nor full column-rank, the full rank decomposition is utilized for calculating $\mathbf{J}_f^{\#T}$ Doty, Melchiorri, and Bonivento (1993). The full recovery of the lost wrench after actuator failures requires that all components of the mobile platform task mapped onto the orthogonal complement of the range space of $\mathbf{J}_f^T$ be zero, i.e., $(\mathbf{I} - \mathbf{J}_f^T\mathbf{J}_f^{\#T})\mathbf{F} = \mathbf{0}$ Notash (2011a).

3 Recoverable and Non-Recoverable Subtasks

In this section, the mobile platform task is partitioned into the recoverable and non-recoverable subtasks. That is, the components of the mobile platform wrench that can be fully recovered and those that cannot be fully recovered are identified in order to compensate for the recoverable components of the lost wrench due to the actuator failure and optimize the secondary criterion. Substituting equation (4) into equation (3) results in

$$\begin{aligned}\Delta\mathbf{F}_r &= \mathbf{J}^T(\boldsymbol{\tau}-\boldsymbol{\tau}_f) - \mathbf{J}_f^T\Delta\boldsymbol{\tau}_{corr} = (\mathbf{I}-\mathbf{J}_f^T\mathbf{J}_f^{\#T})\mathbf{J}^T(\boldsymbol{\tau}-\boldsymbol{\tau}_f)\\ &= (\mathbf{I}-\mathbf{J}_f^T\mathbf{J}_f^{\#T})\Delta\mathbf{F}_f \end{aligned} \tag{6}$$

The full recovery of the manipulator is achieved if either the lost wrench belongs to the range space of $\mathbf{J}_f^T$ in order that $(\mathbf{I}-\mathbf{J}_f^T\mathbf{J}_f^{\#T})\Delta\mathbf{F}_f = \mathbf{0}$ or $\mathbf{J}_f^T$ is full row-rank so as $(\mathbf{I}-\mathbf{J}_f^T\mathbf{J}_f^{\#T}) = \mathbf{0}$. When the lost wrench is mapped onto the orthogonal complement of the range space of $\mathbf{J}_f^T$ or $(\mathbf{I}-\mathbf{J}_f^T\mathbf{J}_f^{\#T}) \neq \mathbf{0}$, the partial recovery is obtained and not all components of the mobile platform task will be recovered. Once the failure occurs, the task is broken down into the recoverable and non-recoverable subtasks. The m_1 entries of $\mathbf{F}$, i.e., $\mathbf{F}_1$, are identified as recoverable subtasks when the corresponding lost wrench $\Delta\mathbf{F}_{1f} \in \mathbb{R}^{m_1\times 1}$ is projected onto the range space of the reduced Jacobian matrix, $\mathbf{J}_{1f}^T \in \mathbb{R}^{m_1\times n}$ or $(\mathbf{I}-\mathbf{J}_{1f}^T\mathbf{J}_{1f}^{\#T}) = \mathbf{0}$. The remaining m_2 entries are taken as non-recoverable subtasks, $\mathbf{F}_2$.These conditions are valid provided the overall forces/torques do not exceed the limit of the healthy joints. All the combinations of $1, 2, \ldots, 6$ components of the wrench are checked to identify which combination(s) is fully recovered. Therefore, $\sum_{i=1}^{6}\binom{6}{i} = 63$ combinations need to be checked for an n DOF, $n \geq 6$, spatial manipulator. The value of m_1 is associated with the maximum number of recovered components of the wrench. If there are more than one m_1 combinations and the wrench components have different priority, there will be one solution for the desired combination. In the case that some of the components have equal importance, the combination that produces the minimum value of the secondary goal is selected. Considering recoverable subtasks, $\mathbf{F}_1$, and non-recoverable subtasks, $\mathbf{F}_2$ and $\mathbf{F} = \begin{bmatrix}\mathbf{F}_1 & \mathbf{F}_2\end{bmatrix}^T$, the static force equation could be rearranged as equation (7)

$$\begin{bmatrix}\mathbf{F}_1\\ \mathbf{F}_2\end{bmatrix} = \begin{bmatrix}\mathbf{J}_1^T\\ \mathbf{J}_2^T\end{bmatrix}\boldsymbol{\tau} \tag{7}$$

where $\boldsymbol{\tau}$ contains the force/torque of active joints and $m_1 + m_2 = 6$. For a given joint forces/torques, the wrench applied by the mobile platform is cal-

culated using equation (7). However, in some applications, the mobile platform wrench is known and the problem is finding the joint forces/torques. In this paper, $\mathbf{J}_1^T$is used to provide the minimum 2-norm for the vector of the joint forces/torques $\boldsymbol{\tau} = \mathbf{J}_1^{\#T}\mathbf{F}_1$.The relation between the joint force/torque vectors after and before the failure is

$$\boldsymbol{\tau}_f = \mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\boldsymbol{\tau} + \boldsymbol{\tau}_{ch} \tag{8}$$

where $\boldsymbol{\tau}_{ch}$ is an $n \times 1$ vector with zero entries for the healthy active joints and τ_{ch} for the entries associated with the failed active joints.

When the actuator of joint h is failed, provided that the lost wrench is not in the range space of $\mathbf{J}_{1f}^T$, the correctional force/torque is applied on the healthy joints to provide the required mobile platform task. Therefore equation (4) can be rewritten as

$$\begin{aligned}\Delta\boldsymbol{\tau}_{corr} &= \mathbf{J}_{1f}^{\#T}\mathbf{J}_1^T(\boldsymbol{\tau} - \boldsymbol{\tau}_f) + (\mathbf{I} - \mathbf{J}_{1f}^{\#T}\mathbf{J}_{1f}^T)\mathbf{k}_1 \\ &= \mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\boldsymbol{\tau} + (\mathbf{I} - \mathbf{J}_{1f}^{\#T}\mathbf{J}_{1f}^T)\mathbf{k}_1\end{aligned} \tag{9}$$

In this paper, the secondary goal is to minimize the 2-norm of the vector of the non- recoverable subtasks and the 2-norm of the overall force/torque vector of the healthy joints. The secondary goal is expressed as

$$\|\mathbf{W}_1(\mathbf{F}_2 - \mathbf{F}_{2r})\|_2^2 + \|\mathbf{W}_2(\boldsymbol{\tau}_f + \Delta\boldsymbol{\tau}_{corr})\|_2^2 \tag{10}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are non-singular weighting matrices. When the failure arises, the force/torque of the healthy joints is adjusted such that the secondary goal is optimally accomplished by minimizing the secondary criterion (10) without degrading the recovered wrench capability. Considering $\mathbf{F}_{2r} = \mathbf{J}_2^T\boldsymbol{\tau}_f + \mathbf{J}_{2f}^T\Delta\boldsymbol{\tau}_{corr}$, equation (8) and equation (9)

$$\begin{aligned}\begin{bmatrix}\mathbf{W}_1(\mathbf{F}_{2r} - \mathbf{F}_2) \\ \mathbf{W}_2(\boldsymbol{\tau}_f + \Delta\boldsymbol{\tau}_{corr})\end{bmatrix} = &\begin{bmatrix}\mathbf{W}_1\mathbf{J}_2^T\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_1\mathbf{J}_{2f}^T\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T} \\ \mathbf{W}_2\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_2\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T}\end{bmatrix}\mathbf{F}_1 + \\ &\begin{bmatrix}\mathbf{W}_1\mathbf{J}_2^T \\ \mathbf{W}_2\end{bmatrix}\boldsymbol{\tau}_{ch} + \begin{bmatrix}\mathbf{W}_1\mathbf{J}_{2f}^T\tilde{\mathbf{J}}_{1f}^T \\ \mathbf{W}_2\tilde{\mathbf{J}}_{1f}^T\end{bmatrix}\mathbf{k}_1 + \begin{bmatrix}-\mathbf{W}_1 \\ \mathbf{0}\end{bmatrix}\mathbf{F}_2\end{aligned} \tag{11}$$

where $(\mathbf{I} - \mathbf{J}_{1f}^{\#T}\mathbf{J}_{1f}^T) = \tilde{\mathbf{J}}_{1f}^T$ .It is obvious that expression (10) can be equivalently written as $\left\|\begin{matrix}\mathbf{W}_1(\mathbf{F}_2 - \mathbf{F}_{2r}) \\ \mathbf{W}_2(\boldsymbol{\tau}_f + \Delta\boldsymbol{\tau}_{corr})\end{matrix}\right\|_2^2$. Since $\begin{bmatrix}\mathbf{W}_1\mathbf{J}_{2f}^T\tilde{\mathbf{J}}_{1f}^T \\ \mathbf{W}_2\tilde{\mathbf{J}}_{1f}^T\end{bmatrix}$ has full column

rank, expression (10) is minimized if and only if

$$\mathbf{k}_1 = \begin{bmatrix} \mathbf{W}_1\mathbf{J}_{2f}^T\tilde{\mathbf{J}}_{1f}^T \\ \mathbf{W}_2\tilde{\mathbf{J}}_{1f}^T \end{bmatrix}^{\#} \left\{ \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{0} \end{bmatrix} \mathbf{F}_2 \right. \tag{12}$$
$$\left. - \begin{bmatrix} \mathbf{W}_1\mathbf{J}_2^T\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_1\mathbf{J}_{2f}^T\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T} \\ \mathbf{W}_2\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_2\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T} \end{bmatrix} \mathbf{F}_1 - \begin{bmatrix} \mathbf{W}_1\mathbf{J}_2^T \\ \mathbf{W}_2 \end{bmatrix} \boldsymbol{\tau}_{ch} \right\}$$

Substituting equation (12) into equation (9) gives

$$\Delta\boldsymbol{\tau}_{corr} = \mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T}\mathbf{F}_1 + \tilde{\mathbf{J}}_{1f}^T \begin{bmatrix} \mathbf{W}_1\mathbf{J}_{2f}^T\tilde{\mathbf{J}}_{1f}^T \\ \mathbf{W}_2\tilde{\mathbf{J}}_{1f}^T \end{bmatrix}^{\#} \left\{ \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{0} \end{bmatrix} \mathbf{F}_2 \right. \tag{13}$$
$$\left. - \begin{bmatrix} \mathbf{W}_1\mathbf{J}_2^T\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_1\mathbf{J}_{2f}^T\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T} \\ \mathbf{W}_2\mathbf{J}_1^{\#T}\mathbf{J}_{1f}^T\mathbf{J}_1^{\#T} + \mathbf{W}_2\mathbf{J}_{1f}^{\#T}\Delta\mathbf{J}_1^T\mathbf{J}_1^{\#T} \end{bmatrix} \mathbf{F}_1 - \begin{bmatrix} \mathbf{W}_1\mathbf{J}_2^T \\ \mathbf{W}_2 \end{bmatrix} \boldsymbol{\tau}_{ch} \right\}$$

The correctional force/torque derived in equation (13) leads to $\Delta\mathbf{F}_{1r} = \mathbf{0}$ and at the same time minimizes the error of the non-recoverable wrench and the overall active joint force/torque jump when the failure occurs.

4 Case Study: 6-DOF Stewart-Gough Manipulator

In this paper, a 6-DOF Stewart-Gough manipulator depicted in Figure 1 is used as a case study. In this manipulator, six identical legs connect the mobile platform to the base platform by spherical joints at points B_i and A_i, $i = 1, 2, \ldots, 6$, respectively. Each leg consists of an upper link and a lower link connected by a prismatic joint. The position vector of the center of the mobile platform, point P, in the base frame is $\mathbf{p} = [p_x\ p_y\ p_z]^T$. The transpose of the Jacobian matrix relating the force of the prismatic joints to the mobile platform wrench is reported in Tsai (1999).

The manipulator is used as a force feedback haptic device in a training simulator. Parallel manipulators have higher stiffness and lower inertia compared to serial manipulators. Therefore, their application as an interface provides a good natural transparency in the system that leads to better stability and reproduces required forces and moments Papadopoulos, Vlachos and Mitropoulos (2002). The coordinates of the base attachment points, A_i, $i = 1, \ldots, 6$, in the base frame $\Psi(x, y, z)$ are $[0.24\ 0\ 0]^T$, $[0.16\ 0.16\ 0]^T$, $[-0.16\ 0.16\ 0]^T$, $[-0.24\ 0\ 0]^T$, $[-0.16\ -0.16\ 0]^T$ and $[0.16\ -0.16\ 0]^T$ meters, respectively. The coordinates of connection points B_i on the mobile platform in the moving frame $\Gamma(x', y', z')$ are $[0.18\ 0\ 0]^T$, $[0.12\ 0.12\ 0]^T$, $[-0.12\ 0.12\ 0]^T$, $[-0.18\ 0\ 0]^T$, $[-0.12\ -0.12\ 0]^T$ and $[0.12\ -0.12\ 0]^T$

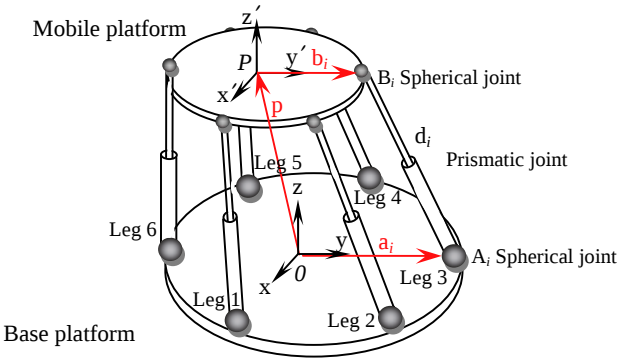


Figure 1. A general Stewart-Gough parallel manipulator.

meters, respectively. The position vector of point P in the base frame is $\mathbf{p} = [-0.04 \;\; 0.03 \;\; 0.81]^T$ meters and the Euler angles are $\psi_x = 30°$, $\theta_y = 20°$ and $\varphi_z = 30°$.

For this manipulator, different cases of failures are analyzed and one case is reported here when actuators of joints 1 and 3 are failed such that $\tau_{c1} = 15$ N and $\tau_{c3} = 10$ N. The desired wrench applied by the mobile platform is $\mathbf{F} = [0.0 \;\; 0.684 \;\; 0.944 \;\; -0.792 \;\; -0.794 \;\; 0.126]^T$ (N and N.m). In this case, the maximum number of the recoverable wrench components is 4, with 15 different combinations for $\mathbf{J}_{1f}$ considering all 4 out of 6 components of wrench. If all components are equally important, the combination including F_y, F_z, M_y and M_z as the recoverable subtasks minimizes the secondary goal among the 15 combinations. For this combination, the force of the actuators before the failure is $\boldsymbol{\tau} = \mathbf{J}_1^{\#T}\mathbf{F}_1 = [7.36 \;\; -3.85 \;\; -3.60 \;\; 5.03 \;\; -2.91 \;\; -0.97]^T$ (N), where using $\boldsymbol{\tau} = \mathbf{J}^{-T}\mathbf{F} = [-0.65 \;\; 5.03 \;\; -5.54 \;\; -2.63 \;\; -6.74 \;\; -2.03]^T$(N).

Due to the failure, the wrench is changed as $\mathbf{F}_f = [0.54 \;\; 21.92 \; -0.41 \;\; 0.63 \;\; -1.91 \;\; 1.60]^T$ (N, N.m). The correctional force applied to the healthy joints is $\Delta\boldsymbol{\tau}_{corr} = [0.0 \; -11.25 \;\; 0.0 \; -9.85 \;\; 0.59 \; -0.68]^T$ (N). Using equation (12), the arbitrary vector $\mathbf{k}_1$ is calculated so as to minimize the secondary goal in expression (10). Adding the correctional force to the force of the healthy joints gives the overall force $\Delta\boldsymbol{\tau}_{corr} + \boldsymbol{\tau}_f = [15 \; -15.11 \;\; 10 \; -4.82 \; -2.32 \; -1.65]^T$ (N). The mobile platform wrench after applying the overall force is $\mathbf{F}_r = [-0.580 \;\; 0.684 \;\; 0.944 \;\; -0.164 \;\; -0.794 \;\; 0.126]^T$ (N, N.m) in which the components $2, 3, 5$ and 6 of the wrench are fully recovered, and the vector of the error of the non-recoverable components is $[0.58 \;\; -0.63]^T$(N, N.m).

5 Conclusion

In this paper, the failure recovery of parallel manipulators due to the loss of force/torque of the active joints was studied. Regarding the number and type of actuator failures, manipulator configuration and task/application purposes, the mobile platform task space was broken down into the recoverable and non-recoverable subtasks. Those components of the lost wrench mapped onto the range space of $\mathbf{J}_{1f}^{T}$ were considered as recoverable subtasks. The force/torque of the healthy joints was adjusted such that the secondary goal, comprising the 2-norm of the error of non- recoverable wrench and the 2-norm of the vector of the overall forces/torques of the healthy joints, was minimized.

Bibliography

R.G. Roberts and A.A. Maciejewski. A local measure of fault tolerance for kinematically redundant manipulators. *IEEE Trans. Rob. and Auto.*, vol. 12, no. 4, pp. 543-553, Aug 1996.

L. Notash. A Methodology for Actuator Failure Recovery in Parallel Manipulators. *Mech. and Mach. Theory*, vol.46, pp.454-465, 2011.

L. Notash. Motion Recovery after Joint Failure in Parallel Manipulators. *CCToMM Symp. Mech., Mach., and Mechatronics.*, Montreal, June 2011.

R.S. Jamisola, Jr, A.A. Maciejewski and R.G. Roberts. Failure tolerant path planning for the PA-10 robot operating amongst obstacles. *IEEE Int. Conf. Rob. and Auto.*, vol. 5, pp. 4995-5000, New Orleans, Apr 2004.

R.S. Jamisola, Jr, A.A. Maciejewski and R.G. Roberts. A Path Planning Strategy for Kinematically Redundant Manipulators Anticipating Joint Failures in the Presence of Obstacles. *IEEE/RSJ Int. Conf. Intelligent Rob. and Syst.*, vol.1, pp. 142-148, Las Vegas, Oct 2003.

T. Yoshikawa. Analysis and Control of Robot Manipulators with Redundancy. *The* 1^{st} *Int. Symp. Rob. Res.*, ed. M. Brady and R. Paul, MIT Press, pp.735-747, 1984.

Y. Chen, J.E. McInroy and Y.Yi. Optimal, Fault-Tolerant Mappings to Achieve Secondary Goals without Compromising Primary Performance. *IEEE Trans. Rob. and Auto.*, vol.19, no.4, pp. 680-691, Aug 2003.

L.W. Tsai. Robot Analysis: The Mechanics of Serial and Parallel Manipulators. *John Wiley & Sons*, New York, 1999.

K.L. Doty, C. Melchiorri, and C. Bonivento. A theory of generalized inverse applied to robotics. *Int. J. Rob. Res.*, vol. 12, no. 1, pp.119, 1993.

E. Papadopoulos, K. Vlachos, D. Mitropoulos. Design of a 5-dof haptic simulator for urological operations. *IEEE Int. Conf. Rob. and Auto.*, vol.2, pp. 2079-2084, Washington, Aug 2002.

Vibration control of an industrial robot with a flexible arm using IDCS

Kengo AOKI , Gentiane VENTURE and Yasutaka TAGAWA
Tokyo University of Agriculture and Technology

Abstract We propose a fast motion controller for a robot which has a flexible arm using IDCS control scheme. We test IDCS with the controller suitable for vibration suppressing: DMM to the lift table and consider the target tracking of IDCS with suppressing the vibration of the flexible board.

1 Introduction

Robots are expected to spread in various environments and required to achieve equally fast, precise and safe movements during various tasks. With the increased complexity of the tasks, these robots have complex structures, yet simple controllers. However, such simple controllers cannot be used with these robots as they cannot satisfy simultaneously all the required performances. And to be fast and accurate they often become unstable when using high gain. Therefore, we need control techniques that can guaranty high performances: fast response, low over-shoot and stability. We applied in simulation and actually the IDCS controller as in Ji and et al. (2009), Tagawa and et al. (2010), Tagawa and Fukui (1994), to a manipulator robot to consider the validity of IDCS. We already applied the most common controller used industrial robots: PID controller to IDCS in simulation and verified that the performances of IDCS is superior to PID Aoki and et al. (2011). Nevertheless, the performance of IDCS with PID is not good regarding vibration suppression. This is due to the performance of PID that is not suitable for vibration suppression. In fact, IDCS can suppress the vibration with the proper controller: DMM Y.Tagawa and et al. (2009), Tagami and et al. (2008).

2 The Proposed Control Scheme

IDCS, Inverse Dynamics Compensation via 'Simulation of feedback control systems', consists in obtaining numerically the inverse dynamics via

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_27,

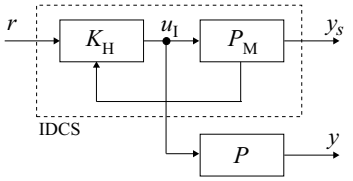


Figure 1. Concept of IDCS control scheme

feedback control simulation, using only the forward dynamic model of the system. When controlling a real system, unwanted factors such as unknown dynamics of the plant and model parameter errors occur, and degrade the controller performances. However in simulation excellent results can be obtained since the environment is completely free from the above undesirable factors. Consequently, the controller can be used with its best performances in simulation. IDCS achieves this goal by using the simulated environment to generate the controller output u_I as in Fig. 1. Where P is a system to control y is the output of the system. The dotted box encloses the simulation environment, where P_M is the forward model of the plant P. Thanks to the simulation environment a high-performance controller K_H is used, and the output y_s of P_M is close to the desired reference r. Since P_M is the precise model of the real system P, when u is input to P, the corresponding output y of the system is close to r. Therefore, the output y of the system P, follows the target r. The dotted frame corresponds exactly to the computation of P^{-1}. It has the same function as the computation of the inverse dynamics of P. IDCS can also perform ideal control with less sensors.

3 The Controlled Object and Modeling

3.1 The controlled object

We show the controlled object in Fig. 2. The controlled object is a robot for transporting wafers which consists in a lifting table and robot arm for the task. Hence we need to control the robot with enough precision and velocity, without leading to unstability. For our experiments, to represent the grasping tool with a wafer, we fix a flexible board with a mass at the extremity of the robot arm to consider the validity of vibration suppression control with IDCS. The controlled object is composed of four joints: one active prismatic joint R_1, one active revolute joint θ_2, two passive revolute joint θ_3, θ_4. The motion of θ_3 and θ_4 is constrained to θ_2 through gear structure written in Eq. 1.

$$\theta_2 = -\frac{1}{2}\theta_3 = \theta_4 \tag{1}$$

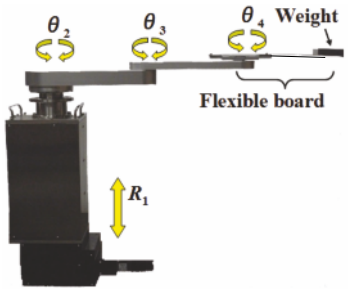


Figure 2. The whole system controlled

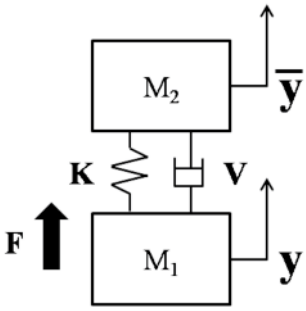


Figure 3. Geometric structure of the controlled object for DMM and IDCS application

We can measure the displacement and angle of of each active joint with the encoders. It is also possible to get information on the motion of the flexible board from the acceleration sensor placed at its tip. In order to consider the viscoelasticity of the flexible board, we considered the end of the flexible board as a virtual prismatic joint. In this condition, we use the Denavit Hertenberg(DH) parameters to generate the kinematics and dynamics (forward and inverse) of the controlled object by using a symbolical software package called SYMORO+ Khalil and Creusot (1997).

3.2 Estimate the transfer function

We estimate the transfer function from the forward dynamic model based on spectrum analysis to design DMM controller. To consider only the motion of the lift table, the geometric structure of the controlled object is shown as Fig. 3. Where we get the parameters such as the mass by means of identifying the inertial parameters using generalized coordinates of the baselink, the joint angles and the external forces informationAoki and et al. (2011). M_1,M_2 mean the mass hold up by the lift table and flexible board, K and V each mean the viscoelasticity of the flexible board, F means the force of the lift table. From Fig. 3, the motion equation are shown as Eq. 2 and Eq. 3. From Fig. 3, the transfer function P_{Fy} from the force F to the displacement of the lift table y=y_1 is given by Eq. 4 and the frequency response is shown in Fig.4. The transfer function $P_{F\bar{y}}$ from the force of the lift table F to the acceleration of the tip of the flexible board $\bar{y}$=$\ddot{y_2}$ is also given by Eq. 5 and the frequency response is shown as Fig.5.

$$M_1\ddot{y_1} = V(\dot{y_2} - \dot{y_1}) + K(y_2 - y_1) \tag{2}$$

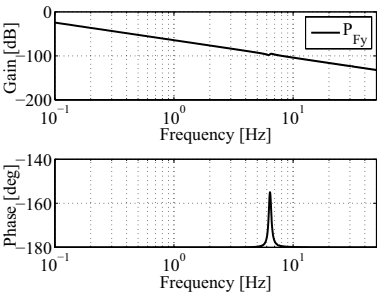


Figure 4. Frequency response P_{Fy}

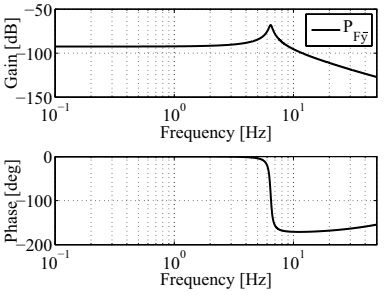


Figure 5. Frequency response $P_{F\bar{y}}$

$$M_2\ddot{y_2} = -V(\dot{y_2} - \dot{y_1}) - K(y_2 - y_1) \tag{3}$$

$$P_{Fy} = \frac{M_2 s^2 + Vs + K}{M_1 M_2 s^4 + (M_1 + M_2)Vs^3 + (M_1 + M_2)Ks^2} \tag{4}$$

$$P_{F\bar{y}} = \frac{Vs + K}{M_1 M_2 s^2 + (M_1 + M_2)Vs + (M_1 + M_2)K} \tag{5}$$

Then, we also consider the frequency response from input voltage u to the force of the lift table F to get the transfer function P_{uF}. We make the experiment that input sine wave sweep to the input voltage with the acceleration sensor placed at it. In this case, we can approximate the transfer function from input voltage to the acceleration of the lift table as second-order system and P_{uF} is given by Eq. 6.

$$P_{uF} = \frac{M_1 b_1 s^2}{s^2 + a_1 s + a_2} \tag{6}$$

Above all, from Eq. 4, Eq. 5, Eq. 6, the transfer function from u to y and $\bar{y}$ are given by Eq. 7, Eq. 8.

$$P_{uy} = \frac{d_2 s^2 + d_1 s + d_0}{s^4 + c_3 s^3 + c_2 s^2 + c_1 s + c_0} \tag{7}$$

$$P_{u\bar{y}} = \frac{e_3 s^3 + e_2 s^2}{s^4 + c_3 s^3 + c_2 s^2 + c_1 s + c_0} \tag{8}$$

4 Controller

DMM controller (Dual Model Matching), is a CTFM-based control system Y.Tagawa and et al. (2009), Tagami and et al. (2008). DMM is 2 DOF

control system with the feedforward controller and feedback controller which allows the closed transfer characteristic from target r, disturbance q and noise v to output y to be determined as we choose. As shown in Fig. 6, the feedforward controller C_{rw} and feedback controller C_{hw} are given by Eq. 9. Where they have to meet Eq. 10 and W_{ij} is the closed loop transfer function from i to j. As shown in Fig. 6 and Eq. 9, W_{vu}, W_{ru} and W_{qu}, are each related to robust stability performance, tracking performance and disturbance suppression.

$$\begin{bmatrix} C_{rw} & C_{hw} \end{bmatrix} = \begin{bmatrix} W_{qu}^{-1}W_{ru} & W_{qu}^{-1}W_{vu} \end{bmatrix} \tag{9}$$

$$W_{qu} = 1 + W_{vu}P_{uy} \tag{10}$$

We consider the block diagram of the controlled object as Fig. 7. It is a SIMO system. From Fig. 7, each controller is given by Eq. 11.

$$\begin{aligned} C_{rw} &= W_{ru}(1 + W_{vu}P_{uy} + W_{\bar{v}u}P_{u\bar{y}})^{-1} \\ C_{vw} &= W_{vu}(1 + W_{vu}P_{uy} + W_{\bar{v}u}P_{u\bar{y}})^{-1} \\ C_{\bar{v}w} &= W_{\bar{v}u}(1 + W_{vu}P_{uy} + W_{\bar{v}u}P_{u\bar{y}})^{-1} \end{aligned} \tag{11}$$

In Eq. 9, DMM tracking characteristics are chosen: (1) depression of first-order mode, (2) noise reduction. Then, the frequency response of the plant and DMM combined are as shown in Fig. 8, Fig. 9. From Fig. 8, Fig. 9, we verify the suppression of the resonance frequency of the flexible board.

5 Simulation

We now run a simulation to move the lift table of the controlled object to the target position, the vibration of the flexible board using the obtained IDCS and DMM controller must be suppressed. We compare the performances of

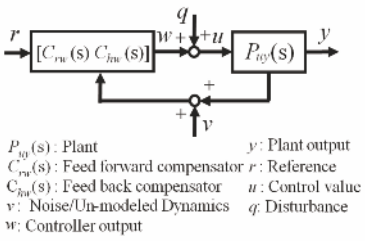


Figure 6. Control synthesis by DMM

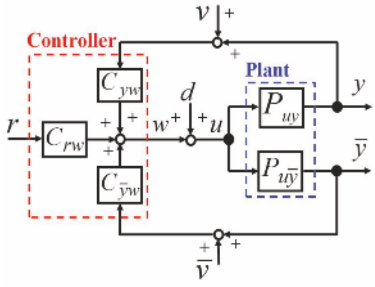


Figure 7. Controller design of DMM

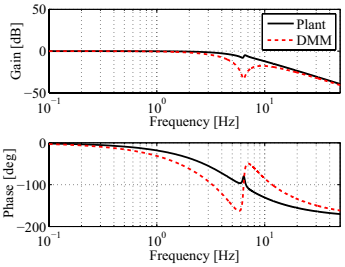


Figure 8. Frequency response of W_{ry}

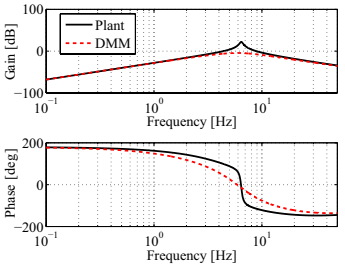


Figure 9. Frequency response of $W_{r\bar{y}}$

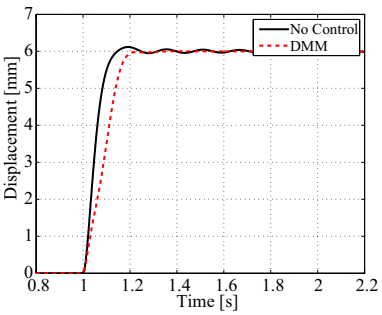


Figure 10. Comparison of No control, IDCS in simulation

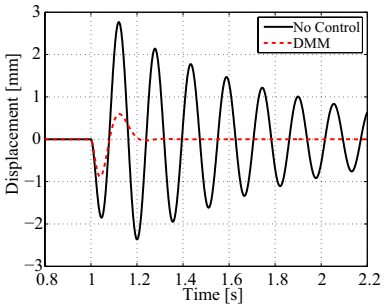


Figure 11. Displacement of the end of flexible board by No control and IDCS in simulation

IDCS controller with no control to consider the validity of IDCS. We input a target displacement r into the lifting table. Considering viscoelasticity of the flexible board, we measure the acceleration of the end of flexible board as $\bar{y}$. Firstly, we input a step signal reaching 6 mm, into r to obtain the simulation result shown in Fig. 10. The displacement calculated from the acceleration of the end of flexible board $\bar{y}$ is also shown in Fig. 11. From Fig. 10, compared with no control, the performance of IDCS gives an excellent target tracking and stability. The vibration in IDCS is also smaller. Above all, we have verified in simulation that the target tracking and stability of IDCS and the suppression of the vibration.

6 Experiment

We test experimentally the proposed controller such as in the simulation. Fig. 12 and Fig. 13 show the results. As shown in Fig. 12, compared with DMM, the performance of IDCS have virtually the same effect. The

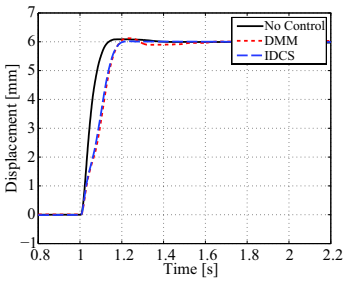


Figure 12. Comparison of No control, DMM controller and IDCS in experiment

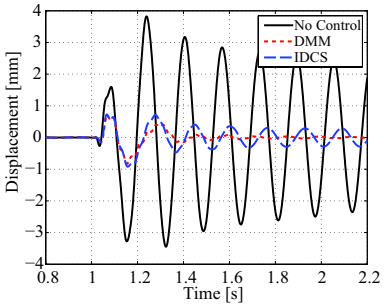


Figure 13. Displacement of the flexible board. No control, DMM controller and IDCS in experiment

tip of the flexible board vibrates as shown Fig. 13. The vibration in IDCS have also the same effect compared with DMM. In both cases, vibration is suppressed more effectively compared to no control case. However, after two seconds, the vibration with IDCS remains although with DMM stop vibrating. This is due to the model errors on the actuator model. Above all, we ascertain the validity of IDCS on the actual robot.
We consider the robustness of IDCS by increasing the mass of each link of P to achieve a parameter error of 30% between P and P_M in Fig. 1. DMM leads the robot unstable due to the error; IDCS performs stably without physical feedback loop. The comparison between IDCS and with mass errors is shown in Fig. 14, Fig. 15. The performance of IDCS has the same effect with or without mass errors. The flexible board vibrates as shown in Fig. 14. The vibration in IDCS with mass errors is larger than IDCS without error but still stable; and it is suppressed better than no control. Therefore, we can certain the robustness of IDCS to the actual robot.

7 Conclusion

We have applied IDCS to a manipulator robot. With multibody dynamics modeling using the DH notations we obtain the forward dynamics model. To get DMM controller, we applied the transfer function estimation. We also tested the validity of IDCS in simulation by comparing the performances of IDCS to no control. We next tested the validity, and IDCS performs with good target tracking and stability and suppression of the vibration. At the end we certain the robustness by comparing IDCS and with mass error. Our results are obtained with constant inertial and mass parameters of robots, however they need to fit the actual condition that fluctuate with time and

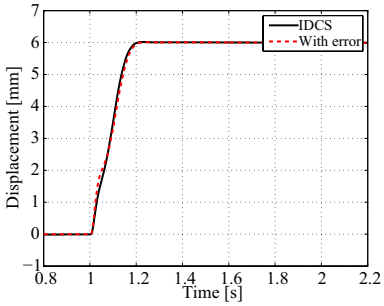


Figure 14. Comparison of IDCS and with errors

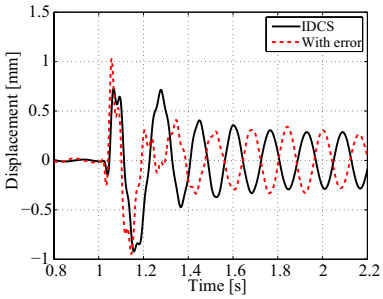


Figure 15. Displacement of the flexible board by IDCS and with errors

task. Therefore IDCS with real time modeling will then be perform in our future work. It also can be interesting to degrade the model by adding more errors, and uncertainties to investigate the robustness of IDCS.

Bibliography

K. Aoki and et al. IDCS control of industrial robots using inertial identification based on contact force sensor. In *Proc. IEEE/Int. Conf. on Intelligent Robot System*, pages 15–18, 2011.

X. Ji and et al. A substructure shaking table test for reproduction of earthquake responses of high-rise buildings. *Earthquake Engng Struct. Dyn.*, pages 1381–1399, 2009.

W. Khalil and D. Creusot. Symoro+: a system for the symbolic modelling of robots. *Robotica*, 15:153–161, 1997.

M. Tagami and et al. Development of an active vibration-canceling system using inertial force generators. In *Int. Conf. Motion and Vibration Control*, pages 15–18, 2008.

Y. Tagawa and et al. Inverse dynamics compensation via simulation of feedback control systems(IDCS). *J. of Sys. and Control Eng.*, 225:137–153, 2010.

Y. Tagawa and K. Fukui. Inverse dynamics calculation of nonlinear model using low sensitivity compensator. In *Dynamics and Design Conference*, pages 185–188, Japan, 1994.

Y.Tagawa and et al. Characteristic transfer function matrix-based linear feedback control system analysis and synthesis. *Int. J. of Control*, 82: 585–602, 2009.

Predictive Functional Control System for Stroke Control of a Pneumatic Tendon-driven Balloon Actuator

Jun-ya Nagase [*] , Toshiyuki Satoh [†]
Norihiko Saga[‡] and Koichi Suzumori [§]

[*] Department of Mechanical and Systems Engineering, Ryukoku University, Shiga, Japan
[†] Faculty of Systems Science and Technology, Akita Prefectural University, Akita, Japan
[‡] Department of Human System Interaction, Kwansei Gakuin University, Hyogo, Japan
[§] Division of Manufacturing System and Physical Science, Okayama University, Okayama, Japan

Abstract In recent years, Japanese society has been aging, engendering a labor shortage of young workers. Robots are therefore expected to be useful to perform tasks such as day-to-day work support for elderly people. Consequently, a tendon-driven balloon actuator has been developed for a robot hand in such environments. This study evaluated stroke control characteristics of a balloon actuator using a predictive functional control (PFC) system. Predictive functional control, a model based on predictive control (MPC) schemes, predicts the future outputs of the actual plant over the prediction horizon and computes the control effort over the control horizon at every sampling instance. Herein, PFC control performance of a one-link finger using a pneumatic balloon actuator is reported.

1 Introduction

Recently, because of the progress of rapid aging and the decrease of younger workers in Japan, robots are anticipated for use in nursing care and welfare services, mainly for the performance of rehabilitation and daily domestic tasks. Safety is a particularly important characteristic of robots intended for use in environments with humans. Safety necessitates dexterous, flexible movements similar to those made by humans. Therefore, the impact force has decreased by making them lightweight. Furthermore, robots that use

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_28,

passive transformation elements have been developed(i.e. Suzumori and Iikura et al. (1992)). Under such circumstances, Saga and Nagase et al. (2006) developed a pneumatic tendon-driven balloon actuator (balloon actuator) with a long stroke and a high power ratio. Moreover, it is compact and lightweight. The balloon actuator has been applied to a robot hand by Nagase and Saga et al. (2010).

The robot hand must control the finger joint angle to manipulate an object with dexterity. However, for pneumatic rubber actuators including the balloon actuator, it is difficult to maintain exact control because these actuators have nonlinear properties, which change their characteristics. In this study, predictive functional control (PFC)(by Richalet (1987)), which has high robustness and which has easily tuned control parameters to achieve good control performance, is applied to a balloon actuator. PFC is a model of predictive control (MPC) that was originally developed for use in industry. Generally, the calculated load of MPC is high because it demands solution of a nonlinear optimization problem every sample time. Therefore, MPC is often applied to chemical plants, at which the sample time is long. However, the calculated load of the PFC is low because the control law consists only of the basis function. Furthermore, it is easier to tune the control parameter because there is only one parameter. Moreover because the PFC control system has an internal model, it has high robustness for varying the properties and disturbances. Herein, the PFC control performance for a one-link finger model using the balloon actuator is evaluated.

2 Predictive functional control

In this section, we briefly present an overview of predictive functional control (PFC). Figure 1 portrays the basic concept of PFC. Presuming that the current time is labeled as time step k, then a set-point trajectory can be defined as a command signal that the process output $y_P(k)$ should follow. The value of the set-point trajectory at the current time step is denoted as $c(k)$. Additionally, a reference trajectory denoted by y_R is shown. This trajectory starts at the current process output $y_P(k)$ and defines a desired trajectory along which process output y_P is expected to approach the set-point trajectory. The reference trajectory has a few coincidence points at which the performance index is defined and the process output $y_P(k)$ is expected to coincide with reference trajectory y_R. Three coincidence points are portrayed as an example in Figure 1. The optimal control input trajectory is then computed based on the predicted output. Once a future control input trajectory has been computed, we apply only the first element of the control input trajectory to the process. At the next time step, we

repeat the entire cycle from the definition of the reference trajectory to the application of the first element of the optimal control input trajectory. We designate this mode of control as the receding horizon control.

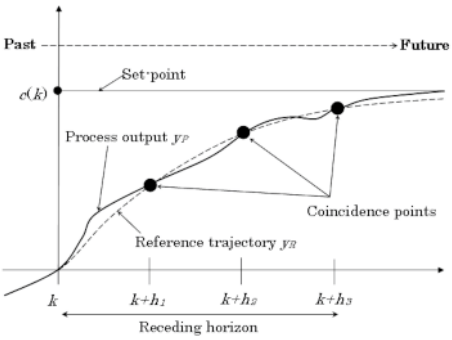


Figure 1. Principle of predictive functional control

Next, we show the basic PFC algorithm using slightly modified formulation for handling time delay (Satoh and Saito et al. (2010)). Now assuming that the plant is stable and has the time delay of L and that the sampling period is T_s, then the development of the PFC algorithm is based on the following SISO discrete-time linear state-space model of the plant:

$$\begin{aligned} x_M(k+1) &= A_M x_M(k) + B_M u(k) \\ y_M(k) &= C_M x_M(k) \end{aligned} \tag{1}$$

where $x_M \in R^n$ stands for the state vector, $u \in R$ signifies the control input, $yM \in R$ denotes the model output. Here, the model output $y_M(k)$ is used to predict the future plant output $y_P(k+d)$ where d is defined as the nearest integer of L/T_s. Then the reference trajectory is defined as presented below:

$$\begin{aligned} y_R(k+d+i) = c(k+d+i) - \alpha^i(c(k+d) - \hat{y}_p(k+d)) \\ i = 0, 1, ... \end{aligned} \tag{2}$$

In that equation $\alpha \in R$ is a parameter that adjusts the approaching ratio of the reference trajectory to the set-point ($0 < \alpha < 1$). Empirically, parameter α has been chosen as $\alpha = e^{-3T_s/T_{CLRT}}$ along with the following three coincidence points as reported by Dieulot and Benhammi et al. (2008).

$$\begin{pmatrix} h_1 & h_2 & h_3 \end{pmatrix}^T = \begin{pmatrix} \frac{T_{CLRT}}{3T_s} & \frac{T_{CLRT}}{2T_s} & \frac{T_{CLRT}}{T_s} \end{pmatrix}^T \tag{3}$$

Therein, T_{CLRT} is called the desired closed-loop response time, which presents the time required for the output response to arrive at 95% of the set-point. In PFC, T_{CLRT} is the only design parameter. Then we can balance the speed of response and the robustness by adjusting T_{CLRT}. In many cases, the performance index is defined as the quadratic sum of the errors between the predicted process output $\hat{y}_P$ and the reference trajectory y_R as follows:

$$D(k) = \sum_{j=1}^{n_h} \{\hat{y}_p(k+d+h_j) - y_R(k+d+h_j)\}^2 \tag{4}$$

where $h_j (j = 0, 1, \ldots, h)$ and n_h respectively denote the coincidence time point and the number of coincidence points. The minimization of the performance equation (4) engenders the optimal control input as

$$\begin{aligned} u(k) = {} & k_0\{c(k+d) - y_p(k)\} + \sum_{m=1}^{d_c} k_m c_m(k+d) \\ & - \sum_{m=1}^{d_e} k_m e_m(k+d) + \tilde{\nu}_x^T x_M(k) + \tilde{\nu}_{xd}^T x_M(k-d) \end{aligned} \tag{5}$$

where d_e and e_m respectively denote the degree of the polynomial approximation and unknown coefficients that are computed based on the basis of the past and present model error. Substances of $\tilde{\nu}_x$ and $\tilde{\nu}_{xd}$ are written in Satoh and Saito et al. (2010).

3 One-link finger model with balloon actuator

3.1 Pneumatic tendon-driven balloon actuator

The pneumatic tendon-driven balloon actuator (balloon actuator) is portrayed in Figure 2. More detailed schematic views are written in Saga and Nagase et al. (2006). The balloon actuator has a high power-to-weight ratio and stroke-to-weight ratio. Therefore, it can generate sufficient stroke and force for the driving robot hand if it is installed in a small space of a robot hand. The balloon actuator is an air-pressure-drive type actuator comprising a silicone tube (balloon) and a tendon wrapped around a tube. The silicone rubber tube is an oval tube with maximum diameter of 21 mm, and minimum diameter of 9 mm, with effective length of 9 mm. The tube is sealed at one end to produce a balloon. Compressed air is supplied through an opening at the other end to expand the balloon. The tendon is made of polypropylene. A nylon fiber sheet is stuck on one side of the tendon to decrease the friction force generated between the tendon and the

balloon. The tube is then fixed at both ends by acrylic plates. The wall is arranged on one side of the balloon to restrain its expansion. To decrease the loss of output efficiency, a roller is arranged in the part where the tendon direction is changed. A basic driving mechanism is the following: the tendon wrapped around the balloon is expanded as the balloon expands, which creates tensile force for the tendon drive.

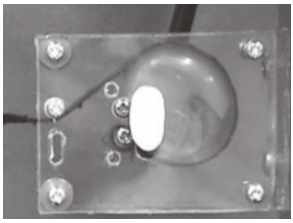


Figure 2. Pneumatic tendon-driven balloon actuator

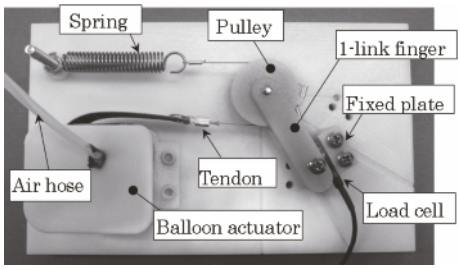


Figure 3. 1-link finger system using balloon actuator

3.2 1-link finger system

In this study, the stroke control performance of the PFC of the balloon actuator is evaluated by conducting a control experiment of the one-link finger system with the balloon actuator. Figure 3 presents a photograph of the one-link finger system used for this study. This finger system consists of the balloon actuator, tension spring, and finger. The finger flexes by restoration force of the tension spring and extends by tensile force of the balloon actuator. Therefore, when this finger system is implemented in a robot hand, this driving mechanism can prevent the fall of an object because the grip force can be secured by the spring if the system halts abnormally. The movable joint angle of the finger is 90 deg, which is almost identical to

the angle used by human fingers. The pulley radius is 11 mm. The tension spring has a spring constant of 1.0 N/mm, with initial force of 5.3 N.

4 Control experiment

4.1 Experimental setup

In this experiment, the compressed air that is input to the balloon is provided by an air compressor [YC-4; Yaezaki Kuatsu K.K.]. The input air pressure is arranged using an electropneumatic regulator [ETR200-1; Koganei Corp]. The finger joint angle is measured using a potentiometer [SV03; Murata Manufacturing Co. Ltd.] installed in the finger joint. The measured angle is input to the PC through the AD/DA board [Q4 PCI board; Quanser] with the sampling time set to 2 ms.

4.2 Model identification

A PFC control system requires the internal model of the plant because it controls it using the predicted future plant output. Therefore, the experiment for the plant model identification is conducted. In the experiment, after the step response characteristic of the finger joint angle for the input pressure is measured, the model parameter is identified by doing curvefitting using the measured data. Consequently, the gain is identified as follows.

$$\theta = 1.10 \times 10^5 p^{3.95} \tag{6}$$

Therein, θ is the joint angle, and p denotes the air pressure to the balloon. The transfer function of the dynamic characteristic is indentified as follows.

$$G(s) = \frac{150}{s^2 + 56s + 150} \tag{7}$$

Because the discrete state space model such as equation (1) is necessary in the PFC control, the continuous transfer function model identified of equation (7) is discretized by Zero-Order Hold, and it is transferred to the state space model

4.3 Experimental method

In the control experiment, the desired finger joint angles are set to 30 deg., 60 deg, 90 deg., respectively, then the control output for the step input is examined. The PFC control parameter is basically only T_{CLRT}. The response speed increases when the T_{CLRT} is decreased because it indicates the desired closed-loop response time. However, in that case, the robust

stability decreases because it is affected easily by noise of the plant output and modeling error. In this study, the smallest T_{CLRT} is used. It is set to 0.10 s so that it does not become a vibrational response.

4.4 Experiment results

Figure 4 presents experimental results for PFC control of the finger. The solid line represents the result of PFC control. The dashed line represents the result of feed-forward control alone.

Regarding comparison of the dynamic characteristics, the feed-forward settling time is 0.70 s, 0.74 s and 0.73 s respectively. That of the PFC is 0.41 s, 0.48 s and 0.73 s. Therefore, the settling time is improved about 40% using PFC when the desired angle is 30 deg. and 60 deg. Next, evaluation of the static characteristics is the following: regarding the feed-forward control, the control output does not converge in the desired value. It increases slowly over time except at 90deg. Apparently, the creep characteristic of the rubber causes such a response. Moreover, when the desired angle is 30 deg, a stick-slip phenomenon is generated, which seems to result from the friction generated between the balloon and the tendon. However, in the case of the PFC, after the control output increases, it continues maintaining the desired value of the joint angle at every desired value. Moreover, the stick-slip phenomenon is not generated at all. Furthermore, then, because the balloon actuator is an air-pressure driven rubber actuator, when the desired value is changed, the control performance might worsen because of effects of the compaction property of the air and the nonlinear property of the rubber. However, the control property of the balloon actuator indicates a stable response at every desired value in this experiment.

5 Conclusions

This study evaluated stroke control characteristics of the balloon actuator by PFC control using a one-link finger system. Our conclusions are the following:

In the stroke control experiment for the step input, in the case of feed-forward control, the control output of the joint angle does not converge to the desired value. However, by application of the PFC control, the settling time is decreased about 40%; then, after the control output of the joint angle increases, it continues to maintain the desired value at every desired value. Moreover the stick-slip phenomenon is not generated at all, although it is generated in the case of feed-forward control.

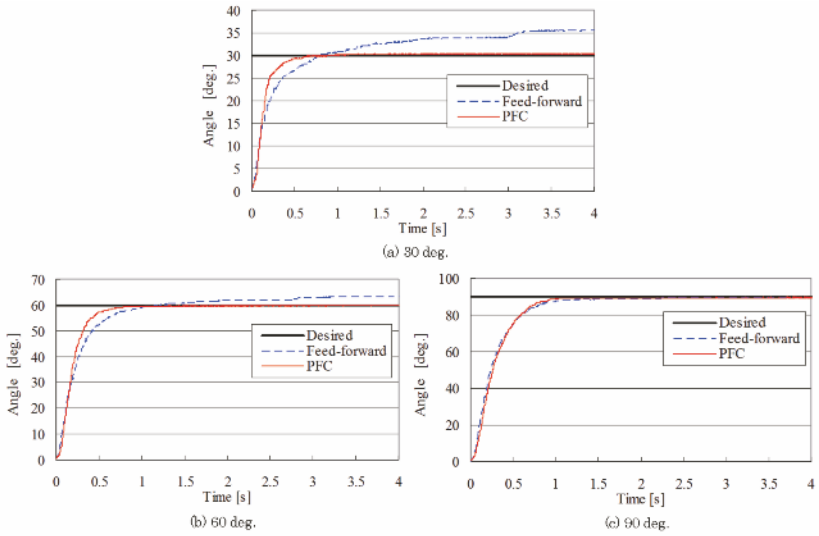


Figure 4. Experiment results

Bibliography

J. Y. Dieulot, T. Benhammi, F. Colas, P. J. Barre, *Composite predictive functional control strategies*, In *A*pplication to positioning axes, International Journal of Computers, Communications and Control, Vol. III, No.1, pages 41–50, 2008.

J. Nagase, N. Saga, N. Saito, *Development of a Tendon Driven Robot Hand Using Pneumatic Balloons*, In *P*roc. of 1st International Conference on Applied Bionics and Biomechanics, pages 47–54, 2010

J. Richalet, S. Abu el Ata-Doss, C. Arber, H.B. Kuntze, A. Jacubasch, W. Schill, *Predictive functional control: application to fast and accurate robot*, In *P*roc. of IFAC 10th World Congress, pages 251-259, 1987.

N. Saga, J. Nagase, Y. Kondo, *Development of a Tendon -Driven System Using a Pneumatic Balloon*, In *J*ournal of Robotics and Mechatronics, Vol.18, No.2, pages 139-145, 2006.

T. Satoh, N. Saito, N. Saga, *Predictive functional control with disturbance observer for pneumatic artificial muscle actuator*, In *P*roc. of the 1st International Conference on Applied Bionics and Biomechanics, 2010.

K. Suzumori, S. Iikura, H. Tanaka, *Applying a Flexible Microactuator to Robotic Mechanisms*, In *I*EEE Control Systems, Vol.12, No.1, pages 21–27, 1992

Redundancy Resolution of the Underactuated Manipulator ACROBOTER

Ambrus Zelei [*], László Bencsik [‡], László L. Kovács [*] and Gábor Stépán [‡]

[*] HAS-BUTE Research Group on Dynamics of Machines and Vehicles, Budapest, Hungary, zelei@mm.bme.hu, kovacs@mm.bme.hu

[‡] Department of Applied Mechanics, Budapest University of Technology and Economics, Budapest, Hungary, bencsik@mm.bme.hu, stepan@mm.bme.hu

Abstract The domestic robot platform ACROBOTER exploits a novel concept of ceiling based locomotion. A climber unit moves on the almost obstacle free ceiling, while carries a swinging unit with a system of suspending and orienting cables. The objective of the robot is the fine positioning of the swinging unit that accomplishes path following or pick and place tasks. Its motion is controlled by ducted fan actuators additionally to the variable length suspending cables. The complexity of the mechanical structure induces the use of natural coordinates for the kinematical description. An algorithm is proposed to control this underactuated and also redundant manipulator, which can be characterized as a control-constraint based computed torque control strategy.

1 Introduction

Recently, more and more robotic systems try to utilize the advantages of the underactuation, like the ones in cases of agile motion and energy efficient operation. The indoor domestic robot called ACROBOTER published by [Stépán and et al. (2009)] is suspended from the ceiling on a cable similarly to cranes, so it is able to utilize the pendulum-like motion efficiently, while the suspending cable also provides power and signal transmission for the swinging unit. The robot utilizes the ceiling that is almost obstacle-free compared to the ground, since it avoids all problems of floor based locomotion caused by randomly placed small objects obstructing the free motion.

The mechanical structure of ACROBOTER can be divided into two main parts; the climber unit (CU) carries the swinging unit (SU), which hangs on a main cable (MC) and three orienting secondary cables (SC) as shown in Fig.1. The CU is a fully actuated, planar RRT robot and is able to move the upper mounting point of the MC in arbitrary location in a plain parallel

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_29,

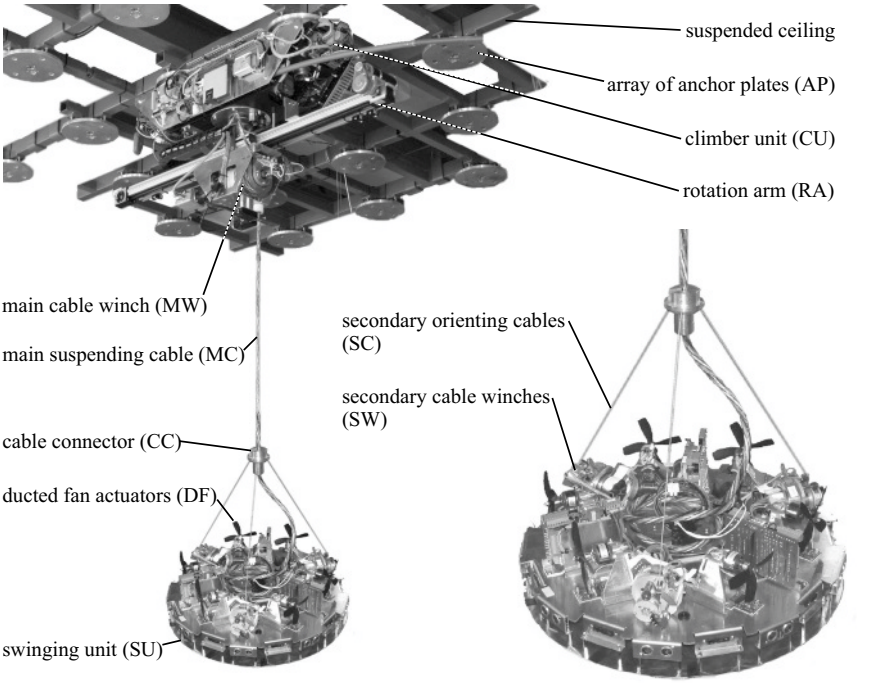


Figure 1. ACROBOTER prototype

to the ceiling. The length of the MC and SCs are varied by servo motors, and the positioning of the SU is assisted by ducted fan actuators. Despite the large number of actuators, the system is still underactuated, because the number of actuators is lower than the DoFs. Additionally, the system is kinematically redundant, because the position and orientation of the SU are prescribed only and there are no requirements for the motion of the CU and the horizontal motion of the cable connector (CC).

We propose a control algorithm for the ACROBOTER system and give a general formalism, which is applicable for underactuated systems with kinematic redundancy, and suits to the principle of natural coordinates.

2 Mechanical model and control task

The planar model of the ACROBOTER manipulator is shown in Fig.2. The CU is substituted by a single horizontal linear drive, the CC is modeled by a

particle with 2 DoFs and the SU is modeled by a rigid body with 3 DoFs. We handle the manipulator as a multibody system described by the Cartesian coordinates of the base points P_L, P_{CC}, P_1 and P_2 of the included bodies. Such set of dependent coordinates is called natural coordinates by [de Jalón and Bayo (1994)]. The vector of the $n = 7$ dependent descriptor coordinates and the $m = 1$ dimensional single geometric constraint representing the constant distance between P_1 and P_2 are introduced as:

$$\mathbf{q} = \left[\begin{array}{ccccccc} x_L & x_{CC} & z_{CC} & x_1 & z_1 & x_2 & z_2 \end{array} \right]^{\mathrm{T}}, \tag{1}$$

$$\boldsymbol{\varphi}(\mathbf{q}) = \left[(x_2 - x_1)^2 + (z_2 - z_1)^2 - (\overline{\mathrm{P_1P_2}})^2 \right]. \tag{2}$$

The system is controlled by $l = 5$ actuators, which is less than the $n - m = 6$ DoFs, thus the system is underactuated. The actuator forces are shown in Fig.2 and are arrayed in the control input vector $\mathbf{u}$:

$$\mathbf{u} = \left[\begin{array}{ccccc} F_L & F_M & F_1 & F_2 & F_T \end{array} \right]^{\mathrm{T}}. \tag{3}$$

The task of the manipulator is to move a specified point of the SU on a prescribed trajectory given by x^d and z^d as functions of time. Besides, the cable connector has to be kept in a given vertical distance h^d_{CC} above the centre of the swinging unit, and the SU has to be kept horizontal. These tasks are expressed by the $g = 4$ dimensional control-constraint vector:

$$\boldsymbol{\gamma}(\mathbf{q}, t) = \left[\begin{array}{c} \frac{z_1+z_2}{2} + h^d_{CC} - z_{CC} \\ \frac{x_1+x_2}{2} - x^d \\ \frac{z_1+z_2}{2} - z^d \\ z_1 - z_2 \end{array} \right]. \tag{4}$$

In general, fully actuated manipulators equipped with more internal DoFs than required to perform a specified task are called kinematically redundant [Spong and Vidyasagar (1989)]. For such systems the inverse kinematic calculation is not unique. The above definition can be specialized for underactuated manipulators, for which the inverse kinematical and dynamical calculations may lead to a unique solution, if the number of independent control inputs and the dimension of the task is equal: $l = g$ [Blajer and Kolodziejczyk (2008); Kovács et al. (2010); Zelei et al. (2011)]. Consequently it is possible to introduce the definition: *underactuated manipulators equipped with more independent control inputs than required to perform a specified task are called dynamically redundant underactuated systems.* This definition is equivalent to kinematic redundancy for fully actuated robots. In contrast, the inverse kinematics of underactuated systems cannot be solved uniquely, so these are always kinematically redundant.

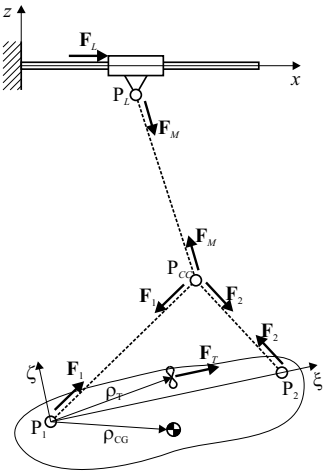


Figure 2. Planar model

	underactuated	kinematically redundant	dynamically redundant
$n-m=g=l$	no	no	no
$n-m>g=l$	yes	yes	no
$n-m>g>l$	yes	yes	yes

Table 1. Redundancy

However, in $g = l$ cases, considering the inverse dynamics, the determination of the control input and kinematics is unique, so dynamic redundancy does not stand. Table 1. summarizes the possible cases for $n - m$ DoFs, g number of specified tasks and l independent actuators.

In the investigated problem $g < l$, thus the system is kinematically and dynamically redundant. One can observe, that in (4) there are no prescriptions for the x_L position of the linear drive. For the redundancy resolution, we use the idea of virtual springs adopted from [McLean and Cameron (1996)]. The angle of the MC is minimized, thus we apply a virtual spring between the x_L and x_{CC} horizontal positions with stiffness k_v. Additionally, the speed of the linear drive should be small, so we introduce a virtual damping element with damping ratio d_v. This optimization rule can be formulated by the $l - g = 1$ dimensional non-holonomic constraint equation $\boldsymbol{\psi}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0}$ with

$$\boldsymbol{\psi}(\mathbf{q}, \dot{\mathbf{q}}, t) \quad = \quad \left[\; d_v \dot{x}_L + k_v (x_L - x_{CC}) \; \right]. \tag{5}$$

The formalism published by [de Jalón and Bayo (1994)] provides the equations of motion for the physical system showed in Fig.2. The dynamical model can be written in the form of a differential-algebraic equation:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}(\mathbf{q})\boldsymbol{\lambda} \quad = \quad \mathbf{Q}(\mathbf{q}) + \mathbf{H}(\mathbf{q})\mathbf{u}, \tag{6}$$

$$\boldsymbol{\varphi}(\mathbf{q}) \quad = \quad \mathbf{0}, \tag{7}$$

where the positive definite mass matrix $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n\times n}$ is constant in case of the use of natural coordinates. However, in the next section we do not focus on this special case, thus, in general the mass matrix may depend on the descriptor coordinates. $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ is the vector of forces arising partly from the dynamics of the system (Coriolis, centrifugal, etc.) and from active forces (springs, dampers, etc.). $\mathbf{Q}(\mathbf{q}) \in \mathbb{R}^n$ is the vector of gravitational forces. $\mathbf{H}(\mathbf{q}) \in \mathbb{R}^{n\times l}$ is the control input matrix and $\mathbf{u} \in \mathbb{R}^l$ is the control input vector. Matrix $\boldsymbol{\varphi}_\mathbf{q}(\mathbf{q}) = \partial\boldsymbol{\varphi}(\mathbf{q})/\partial\mathbf{q} \in \mathbb{R}^{m\times n}$ is the constraint Jacobian associated with the geometric constraints $\boldsymbol{\varphi}(\mathbf{q}) \in \mathbb{R}^m$. $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of the Lagrange multipliers. The dimension l of the control input vector is lower than the DoFs $n - m$, thus the system is underactuated.

3 The control method

The equations of motion (6) and (7) are complemented by the control-constraint equation (also named servo-constraints) and the optimization rule:

$$\boldsymbol{\gamma}(\mathbf{q}, t) = \mathbf{0}, \tag{8}$$

$$\boldsymbol{\psi}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0}. \tag{9}$$

The control-constraint vector can be handled similarly to the geometric constraints (7), however the control-constraints usually depend explicitly on time. We assume that the geometric constraints (7), the control-constraints (8) and the optimization rule (9) are linearly independent and consistent, furthermore (8) and (9) can be satisfied with bounded control forces.

The method of Lagrange multipliers is well known from [de Jalón and Bayo (1994)] regarding the numerical integration of the governing differential-algebraic equation of multibody systems. That method is based on the double time differentiation of the geometric constraints. Similarly, in our work, the geometric constraint equation (7) and the control-constraint equation (8) are formulated at the level of acceleration by differentiating them twice with respect to time, in order to make the acceleration $\ddot{\mathbf{q}}$ appear explicitly:

$$\boldsymbol{\varphi}_\mathbf{q}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_\mathbf{q}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{0}, \tag{10}$$

$$\boldsymbol{\gamma}_\mathbf{q}(\mathbf{q}, t)\ddot{\mathbf{q}} + \dot{\boldsymbol{\gamma}}_\mathbf{q}(\mathbf{q}, \dot{\mathbf{q}}, t)\dot{\mathbf{q}} + \dot{\boldsymbol{\gamma}}_t(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0}, \tag{11}$$

where $\boldsymbol{\gamma}_\mathbf{q}(\mathbf{q}, t) \in \mathbb{R}^{g\times n}$ is the Jacobian of the control-constraint and vector $\boldsymbol{\gamma}_t(\mathbf{q}, t) \in \mathbb{R}^g$ is the partial time derivative of the explicitly time dependent part of the control-constraint. In the application of the method of Lagrange multipliers the geometric constraint equations can be stabilized by the Baumgarte stabilization technique [Baumgarte (1972)]. Similarly,

we extend the acceleration level control-constraint equation (11) as follows:

$$\boldsymbol{\gamma}_{\mathbf{q}}(\mathbf{q},t)\ddot{\mathbf{q}}+\dot{\boldsymbol{\gamma}}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t)\dot{\mathbf{q}}+\dot{\boldsymbol{\gamma}}_{t}(\mathbf{q},\dot{\mathbf{q}},t)+ \\ \mathbf{K}_D[\boldsymbol{\gamma}_{\mathbf{q}}(\mathbf{q},t)\dot{\mathbf{q}}+\boldsymbol{\gamma}_{t}(\mathbf{q},t)]+\mathbf{K}_P\boldsymbol{\gamma}(\mathbf{q},t)=\mathbf{0}. \tag{12}$$

On the contrary, (10) is not stabilized, because the geometric constraints are naturally satisfied. In (12) $\mathbf{K}_P \in \mathbb{R}^{l\times l}$ and $\mathbf{K}_D \in \mathbb{R}^{l\times l}$ are positive definite gain matrices. Equation (12) is asymptotically stable for fixed desired positions [Bencsik and Kovács (2011)].

The optimization rule (9) is formulated by a non-holonomic constraint, hence the acceleration $\ddot{\mathbf{q}}$ appears in its first time derivative:

$$\boldsymbol{\psi}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t)\dot{\mathbf{q}}+\boldsymbol{\psi}_{\dot{\mathbf{q}}}(\mathbf{q},\dot{\mathbf{q}},t)\ddot{\mathbf{q}}+\boldsymbol{\psi}_{t}(\mathbf{q},\dot{\mathbf{q}},t)=\mathbf{0}, \tag{13}$$

where $\boldsymbol{\psi}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t) \in \mathbb{R}^{(l-g)\times n}$ and $\boldsymbol{\psi}_{\dot{\mathbf{q}}}(\mathbf{q},\dot{\mathbf{q}},t) \in \mathbb{R}^{(l-g)\times n}$ are the Jacobian of the optimization rule regarding $\mathbf{q}$ and $\dot{\mathbf{q}}$ respectively. Vector $\boldsymbol{\psi}_t(\mathbf{q},\dot{\mathbf{q}},t)$ is the partial time derivative of the explicitly time dependent part of (9). Since the optimization rule is given in the form of an artificial constraint, (13) has to be stabilized similarly to the control-constraint. We extend (13) with the positive definite gain matrix $\mathbf{K}_\psi$:

$$\boldsymbol{\psi}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t)\dot{\mathbf{q}}+\boldsymbol{\psi}_{\dot{\mathbf{q}}}(\mathbf{q},\dot{\mathbf{q}},t)\ddot{\mathbf{q}}+\boldsymbol{\psi}_{t}(\mathbf{q},\dot{\mathbf{q}},t)+\mathbf{K}_\psi\boldsymbol{\psi}(\mathbf{q},\dot{\mathbf{q}},t)=\mathbf{0}. \tag{14}$$

The unconstrained dynamic equation (6), the acceleration level geometric constraint equation (10), the stabilized acceleration level control-constraint equation (12) and the optimization rule (14) is incorporated in hyper-matrix form as follows:

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}) & \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}(\mathbf{q}) & -\mathbf{H}(\mathbf{q}) \\ \boldsymbol{\varphi}_{\mathbf{q}}(\mathbf{q}) & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\gamma}_{\mathbf{q}}(\mathbf{q},t) & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\psi}_{\dot{\mathbf{q}}}(\mathbf{q},\dot{\mathbf{q}},t) & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \\ \mathbf{u} \end{bmatrix} = \\ \begin{bmatrix} \mathbf{Q}(\mathbf{q})-\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \\ -\dot{\boldsymbol{\varphi}}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} \\ -\dot{\boldsymbol{\gamma}}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t)\dot{\mathbf{q}}-\dot{\boldsymbol{\gamma}}_{t}(\mathbf{q},\dot{\mathbf{q}},t)-\mathbf{K}_D[\boldsymbol{\gamma}_{\mathbf{q}}(\mathbf{q},t)\dot{\mathbf{q}}+\boldsymbol{\gamma}_{t}(\mathbf{q},t)]-\mathbf{K}_P\boldsymbol{\gamma}(\mathbf{q},t) \\ -\boldsymbol{\psi}_{\mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t)\dot{\mathbf{q}}+-\boldsymbol{\psi}_{t}(\mathbf{q},\dot{\mathbf{q}},t)+-\mathbf{K}_\psi\boldsymbol{\psi}(\mathbf{q},\dot{\mathbf{q}},t) \end{bmatrix} \tag{15}$$

from which the control input $\mathbf{u}$, the acceleration $\ddot{\mathbf{q}}$ and the vector of Lagrange multipliers $\boldsymbol{\lambda}$ can be calculated as the function of the measured state $\mathbf{q}$ and $\dot{\mathbf{q}}$ of the system, where $\mathbf{q}$ and $\dot{\mathbf{q}}$ come from the measured values.

4 Simulation results

A round-cornered rectangular shaped trajectory was prescribed, on which the SU was moved along periodically. Fig.3. shows one period of the motion.

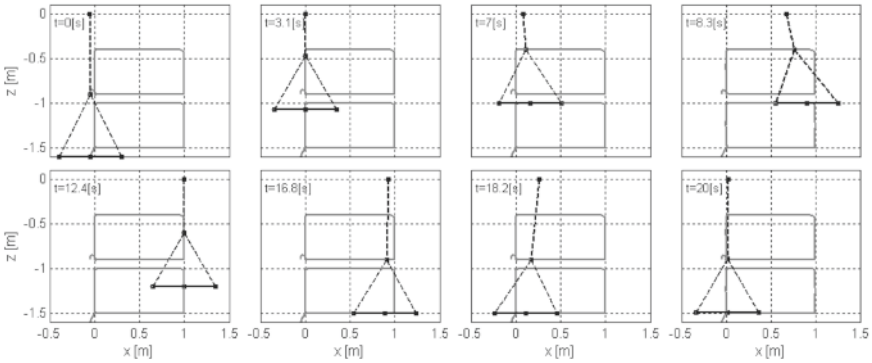


Figure 3. Simulated motion

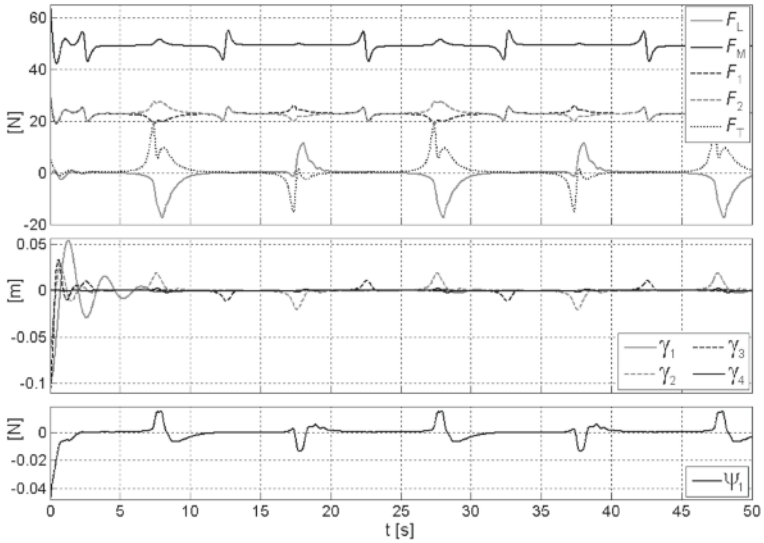


Figure 4. Simulation results: time histories

In Fig.4. the time histories of the control forces, the control-constraint violation and the time history of the violation of the optimization rule can be seen in approximately 2.5 periods. It can be clearly observed that the perturbation applied in the initial time instant is eliminated quickly by the controller, however $\gamma_1 ... \gamma_4$ and ψ_1 values grow up, when larger accelerations are required in the corners of the prescribed trajectory. As a summary of the simulation work, we can say that the violation of the control-constraints

and the optimization rule could be driven to zero in a stable way, and the motion of the manipulator was also stable and smooth.

5 Conclusions

An efficient motion control algorithm was presented and analyzed in case of the cable suspended domestic robot ACROBOTER, which is an underactuated system and also dynamically redundant. A definition of dynamic redundancy was given for underactuated robots. An existing computed torque control algorithm was extended by means of the introduction of an optimization rule in the form of an artificial non-holonomic constraint. The corresponding general formalism was presented, and the numerical simulations of case studies showed the efficiency of the method.

Bibliography

J. Baumgarte. Stabilization of constraints and integrals of motion in dyanamical systems. *Computer methods in applied mechanics and engineering*, 1(1):1–16, 1972.

L. Bencsik and L. L. Kovács. Stability case study of an underactuated service robot. In *Dynamical Systems - Nonlinear Dynamics and Control*, pages 89–94, 2011.

W. Blajer and K. Kolodziejczyk. Modeling of underactuated mechanical systems in partly specified motion. *Journal of Theoretical and Applied Mechanics*, 46(2):383–394, 2008.

J.G. de Jalón and E. Bayo. *Kinematic and dynamic simulation of multibody systems: the real-time challenge*. Springer-Verlag, 1994.

L. L. Kovács, A. Zelei, L. Bencsik, J. Turi, and G. Stépán. Motion control of an under-actuated service robot using natural coordinates. In *Proceedings of ROMANSY 18 Robot Design, Dynamics and Control*, pages 331–338, 2010.

A. McLean and S. Cameron. The virtual springs method: Path planning and collision avoidance for redundant manipulators. *Intl. Journal of Robotics Research*, 15(4):300–319, 1996.

M. W. Spong and M. Vidyasagar. *Robot Dynamics and Control*. John Wiley & Sons, 1989.

G. Stépán and et al. A ceiling based crawling, hoisting and swinging service robot platform. In *Proceedings of Beyond Gray Droids: Domestic Robot Design for the 21st Century Workshop at HCI 2009*, 2009.

A. Zelei, L. L. Kovács, and G. Stépán. Computed torque control of an under-actuated service robot platform modeled by natural coordinates. *Commun Nonlinear Sci Numer Simulat*, 16(5):2205–2217, 2011.

Dexterous Manipulation Planning For an Anthropomorphic Hand

Jean-Philippe Saut*, Anis Sahbani*, Jean-Pierre Gazeau†,
Saïd Zeghloul†, Philippe Bidaud*

* UPMC Univ Paris 06 - CNRS UMR 7222 - ISIR, France
† Institut Pprime - UPR 3346 CNRS - Université de Poitiers - ENSMA, France

Abstract In this paper, we present some results about dexterous manipulation planning with an anthropomorphic hand. The task is to drive the grasped object from a start to a goal configuration. The planning algorithm automatically computes the finger motion and the required regrasping motion i.e. when and how to relocate the contacts on the object's surface. The planner relies on a previously presented method but some extensions were added in order to make it applicable to the hand used in the experiments (a four-fingered hand with a human-inspired kinematics). Some experiments, conducted on the real platform, are presented.

1 Introduction

Many hand designs and mechanical realizations have been proposed in the literature such as the DLR hand (Butterfass et al., 2001), UB-hand (Palli et al., 2011) and the hand of the Shadow robot company. Most of them have been used to validate grasp planners, grasping force computation or control laws. Planning the motion of complex dexterous manipulation tasks has lead to less works and to few experiments. The first theoretic works dealing with dexterous manipulation planning (DMP) have presented description of the configuration space and the kinematic and geometric constraints of the problem (Li et al., 1989; Montana, 1995). Trinkle and Hunter (1991) proposed a solution to plan manipulation tasks with a two-fingered planar hand. The work of Yashima et al. (2003) considers all the possible contact modes (sliding, sliding with rolling, with spinning, etc.) between the fingertips and the object. A global planner explores the configuration space of the object via a rapidly-exploring random tree "RRT" (Lavalle and Kuffner, 2000). From a current configuration of the system hand+object, the planner finds the motion to reach a new object configuration after choosing a set of contact modes at random and after integrating the inverse dynamic

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_30,

model of the system. Xu et al. (2010) describe the problem with the hybrid automaton formalism. A RRT-like algorithm is used to solve the problem. Recently, Bouyarmane and Kheddar (2011) proposed a generalized framework to plan a discrete sequence of multi-contact stances that brings a set of collaborating robots and manipulated objects from a specified initial configuration to a desired goal through non-gaited acyclic contacts with their environment or among each other.
The applications of DMP algorithms to real modern robotic hands are very few. Xue et al. (2008) focused on the particular task of rotating an object. The authors applied their technique on the Schunk anthropomorphic hand (Liu et al., 2008). Experimented were conducted to plan and execute the task of screwing a light bulb. However, the limited manipulation capacities of the hand, that make no stable grasps possible without using the thumb, imposes the bulb to be maintained into a holder.
As previously noted, almost no DMP technique has been applied to real robot hands. In particular, it would be interesting to see the effectiveness of such planner and if theoretic (e.g. different hypotheses) or practical improvements (e.g. online adaptation to compensate drift due to control inaccuracy) are required. The purpose of this paper is to present a complete dexterous manipulation planner and to discuss its performance with a 16-DOFs anthropomorphic hand.
The paper is decomposed as follows. Section 2 recalls the principles of our DMP method and details the extensions added in order to apply it to the specific *LMS hand*. Section 3 gives the description of the aforementioned hand. This hand was developed in the 90's and has so far been successfully used for robotic grasping or simple manipulation tasks. Some trajectories computed by our planner were played on the hand. The results of these experiments are presented in Section 4.

2 Dexterous Manipulation Planning

2.1 Problem Description and Hypotheses

The considered system is composed of n fingers mounted on a fixed palm. Each finger has at least three degrees of freedom (DOF). The manipulated object is rigid and has a known shape. The contacts between the hand and the object can only occur at the fingertips and must be punctual. A configuration is described with the values of the fingers joint angles and the object's pose parameters. A grasp is described as a set of grasping fingers, i.e. fingers in contact with the object, and their respective contact positions on the object's surface. The description of a contact position depends on how the object's surface is represented (e.g. triangle mesh, parametric

surface). We assume the stability of a grasp can be determined from this simple description associated with the contact friction coefficients and object's mass parameters i.e. we consider static stability (see section 2.4).
A *grasp subspace* GS_k is defined as the set of all configurations where the object is grasped with any k fingers. As there can be C_n^k different combinations of grasping fingers, GS_k can be decomposed into the subspaces $(GS_k^i)_{i\in[\![1;C_n^k]\!]}$ where GS_k^i is the space of all grasp with a particular combination of k fingers. In order to be feasible, a computed motion must respect some constraints, listed as follows:

- There must be no collision between bodies, except at contact point between the fingertips and the object.
- The object movements must be induced by the movements of the fingers (i.e. the object cannot move alone).
- The object grasp must be stable.

As in (Saut et al., 2011), we define two kinds of elementary path that satisfy the above constraints: *transfer* path and *regrasping* path. Transfer paths are paths were the object is moved by moving the fingers. The contact positions can change, but only due to the rolling between the fingertip and object surfaces. During a regrasping path, the object is maintained immobile but some fingers break their contact and move freely to go and establish a contact at different positions.

2.2 Method Overview

The multi-fingered manipulation leads to a sequence of finger movements than can involve three- to n-fingered grasps. To each of these grasp sets is associated a grasp subspace. The solutions of a DMP problem consequently belong to the union of all the grasp subspaces.The most exhaustive approach would be to explore this whole union set. However, such an approach would lead to prohibitive resolution time because of the very high dimension of the space to be explored. Instead, we choose to favor the search for a solution inside GS_n. Exploring GS_n will be faster and so will be the resolution of the DMP problem. The construction of the manipulation graph is then decomposed in the following steps:

1. Compute the connected components of GS_n,
2. Determine the connectivity of GS_n components using transfer-regrasping paths (through GS_{n1}),
3. Transform GS_n portions of the solution path into a finite sequence of transfer and regrasping paths.

Figure 1 illustrates the idea for a four-fingered hand. The purpose of step (1) is to develop the graph in GS_n by adding new sample configurations and possibly connecting them to the graph via a local path. Note that the

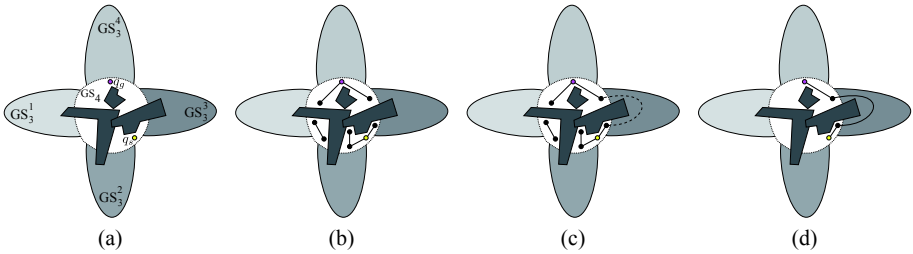


Figure 1. (a) The different subspaces for a 4-fingered hand, the obstacles in $\mathcal{C}$-space (dark polygonal regions) and the start and goal configurations (b) exploration of GS_4 (c) connection of some components of the graph via GS_3 (d) the obtained manipulation path, linking q_s and q_g.

graph is first populated with the start and goal configurations, denoted as q_s and q_g respectively. As possible obstacles in the configuration space may break GS_n into different connected components, it may be necessary to pass through higher-dimensional subspace to link these components. This is the goal of step (2), which tries to link two configurations of the graph, lying in two different components, via a path in GS_{n-1}. Steps (1) and (2) are actually performed alternatively during the planning process. Once q_s and q_g can be linked via a path in the graph, the found path is transformed into a feasible sequence of transfer and regrasping paths (step (3)).
To explore GS_n (step (1)), we consider it to be the configuration space of a single system consisting of the hand grasping the object. Virtual degrees of freedom are added at each contact in order to model the continuity of the grasp (and to allow motion inside GS_n).

2.3 Local Paths Computation

We extend (Saut et al., 2011) by considering the contact rolling during a transfer path since the shape of the fingertips is imposed by the experimental hand. This theoretically induces additional descriptions (positions of the contact on the fingertips) and decisions (where to place the contact on fingertips when computing a grasp). In practice, most of the robotic fingers have only three DOFs like our experiment hand. So having the contact position on the object's surface imposes the configuration of the finger. Compared to fixed contact points, the main difference is the computation of the evolution of a contact point during a transfer path. The general way to solve this problem is to model the contact as an unactuated 3-DOF ball-and-socket joint. The kinematic chain of each finger is then extended

to integrate the contact DOFs (Montana, 1995). In the case of 3-DOFs fingers, the constraint that the contact translational velocity is zero gives:

$$v_o + \omega_o \times r_{c_o} = J_{f_i} \begin{pmatrix} \dot{\theta}_{i,1} \\ \dot{\theta}_{i,2} \\ \dot{\theta}_{i,3} \end{pmatrix} \quad (1)$$

where v_o and ω_o are the object's translational and rotational velocities respectively, r_{c_o} is the contact's position with respect to the object and J_{f_i} is the Jacobian of finger i at the contact point. In our case, J_{f_i} is at most of rank 3 so $(\dot{\theta}_{i,1}, \dot{\theta}_{i,2}, \dot{\theta}_{i,3})$ is fully determined. The relative velocities of object and fingertip can then be computed from $(\dot{\theta}_{i,1}, \dot{\theta}_{i,2}, \dot{\theta}_{i,3})$. Then, if the local parameters of both surfaces are known (curvature, torsion form and scale form at the contact point), it is possible to compute the evolution of the contact point on each surface from the contact equations (Montana, 1995) and consequently adjust the finger configuration. Another solution is to increment the finger joint parameters from Eq.(1) and then shift the fingertip position along the normal to remove the penetration between fingertip and object and compute the new contact position. We chose the latter solution.

2.4 Grasp Stability

As mentioned earlier, the stability is considered from a static-equilibrium point of view. We use the force-closure criterion that guarantees that any external perturbation can be balanced with a set of contact forces. This criterion does not consider limitations on the forces exerted by the fingers. Therefore, it is used during the planning (due to its rapidness), but for each configuration of the final solution trajectory, exerted forces achieving the static equilibrium (i.e. compensating the object's weight) are computed. This was performed by solving an optimization problem using the method proposed in Han et al. (2000). It is then possible to check that the fingers can indeed exert forces with the computed magnitude.

3 The LMS hand

The LMS hand (Daoud et al. (2012), figure 2) is a dexterous hand kinematically inspired by the human hand. Each finger has four actuators but, to remove the system's redundancy, the two last joints are synchronized, leading to coupled articulations like in the human hand. The LMS hand does not have metacarpal articulation and has limited abduction angles, so its possibilities to grasp an object without using the thumb are weak. This constrains the shape of the object to be manipulated and its possible poses

as it will be seen in Section 4.

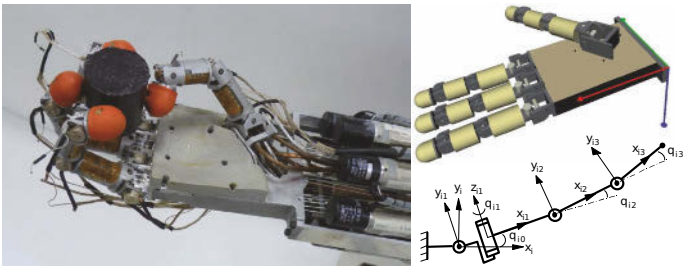


Figure 2. The LMS hand, its CAD model and a finger's kinematic chain.

Each of the 16 joints is driven by a DC actuator located in the forearm. The transmission is ensured by high resistance polyethylene cables. Incremental encoders are placed on the DC motor shafts and on the joints. The difference between the encoder measures gives the cables elongation, which is used to evaluate the grasping forces applied by the fingers to the object, using the neural-network-based method of Daoud et al. (2012).

4 Experimental Results

Two manipulation tasks are proposed to demonstrate the effectiveness of our approach. For each task, a path is first computed off-line in simulation. The planner returns solutions as sequences of key configurations of the fingers. These configurations are the end points of the elementary paths computed by the planner. Configurations between the key configurations can be computed by linear interpolation. Then, the computed path is given as a reference to the robotic hand controller. The choice of the tasks takes into account the limitation of the hand workspace. The first task (figure 3) is the translation of a cylinder (3cm up from its initial configuration). The computation times are of the same order of magnitude of those in (Saut et al., 2011), namely a mean time of 13s (averaged on 100 tests) on a desktop PC for the first task. The second task is the rotation of the same object 45° around its axis of revolution (figure 3).
At the start of an experiment, the hand is set to its start configuration and the object is manually placed in the hand at its start pose by the experimenter. The hand software then computes interpolated configurations from the aforementioned key configurations with a joint angle step of 1° The interpolated configurations are then successively sent to the fingers position controllers. The rotation task and the translation task are composed of 109

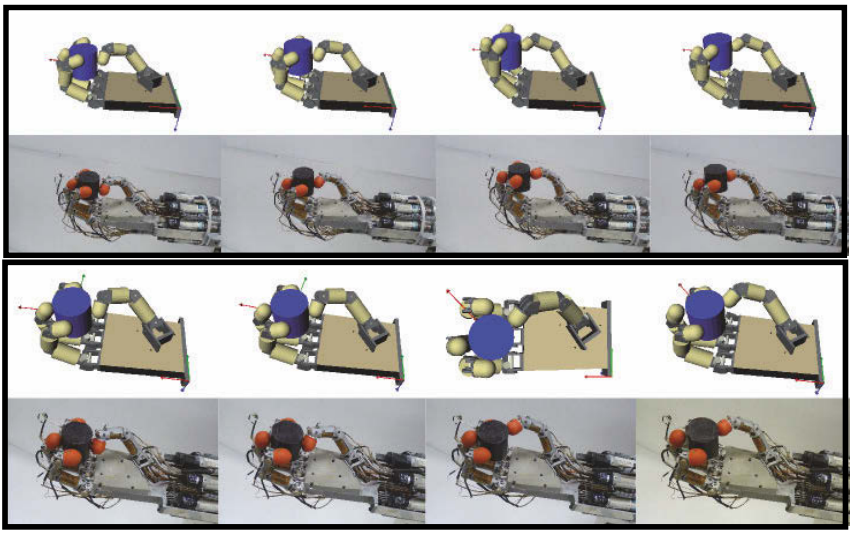


Figure 3. The first task (top block) is to lift up the cylinder (⌀=4cm, length=4cm) from start (left) to goal configuration (right). The second task (bottom block) is to rotate the same object of 45° around its axis.

and 28 key configurations respectively and imply about 17 and 4 contact relocations per finger respectively. The trajectories were successfully executed i.e. the object did not fall from the hand and it reached its final pose. Some steps of the trajectories are shown on figure 3. The error on the final pose of the object is significant (about 0.5cm) but it remains acceptable since there is no external servoing during the manipulation.

5 Conclusion

We have tested a dexterous manipulation planner on a real anthropomorphic hand. The regrasping sequence and the in-between finger motions were planned for different tasks leading to complex manipulation motions with up to dozens of contact relocations per finger. The computed trajectories were given directly as a set of key-points given to the fingers position controllers. The trajectories were successfully executed on the hand. The object did not fall and the error on its final position was moderate for such an open-loop execution. These results suggests that the planner can efficiently plan dexterous manipulation tasks when some shift on the object pose during the execution is acceptable. Further improvements of our dexterous manipu-

lation system could be to add a control loop on the object pose to reduce the shift. Another improvement could be to replan locally the motion if the shift becomes too big to be compensated by a controller.

Bibliography

K. Bouyarmane and A. Kheddar. Multi-contact stances planning for multiple agents. In *IEEE Int. Conf. on Robotics and Automation*, pages 5246–5253, 2011.

J. Butterfass, M. Grebenstein, H. Liu, and G. Hirzinger. Dlr-hand ii: next generation of a dextrous robot hand. In *IEEE Int. Conf. on Robotics and Automation*, pages 109–114, 2001.

N. Daoud, J.P. Gazeau, S. Zeghloul, and M. Arsicault. A real-time strategy for dexterous manipulation: Fingertips motion planning, force sensing and grasp stability. *Robotics and Auton. Systems*, 60(3):377–386, 2012.

L. Han, J.C. Trinkle, and Z.X. Li. Grasp analysis as linear matrix inequality problems. *IEEE Trans. on Rob. and Automation*, 16(6):663–674, 2000.

S. Lavalle and J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In *Algorithmic and Computational Robotics: New Directions*, pages 293–308, 2000.

Z. Li, J.F. Canny, and S.S. Sastry. On motion planning for dexterous manipulation. i. the problem formulation. In *IEEE Int. Conf. on Robotics and Automation*, pages 775–780, 1989.

H. Liu, P. Meusel, G. Hirzinger, M. Jin, Y. Liu, and Z. Xie. The modular multisensory dlr-hit-hand: Hardware and software architecture. *IEEE/ASME Transactions on Mechatronics*, 13(4):461–469, 2008.

D.J. Montana. The kinematics of multi-fingered manipulation. *IEEE Transactions on Robotics and Automation*, 11(4):491–503, 1995.

G. Palli, C. Melchiorri, G. Berselli, and G Vassura. On the design of anthropomorphic dexterous robot hands: The ub hand evolution. *IEEE Robotics and Automation Magazine*, 2011.

J.-P Saut, A. Sahbani, and V. Perdereau. Generic motion planner for robot multi-fingered manipulation. *Advanced Robotics*, 25:23–46, 2011.

J. Trinkle and J. Hunter. A framework for planning dexterous manipulation. In *IEEE Int. Conf. on Rob. and Automation*, pages 1245–1251, 1991.

J. Xu, T.K. Koo, and Z. Li. Sampling-based finger gaits planning for multifingered robotic hand. *Autonomous Robots*, 28:385–402, 2010.

Z. Xue, J.M. Zollner, and R. Dillmann. Dexterous manipulation planning of objects with surface of revolution. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2703–2708, 2008.

M. Yashima, Y. Shiina, and H. Yamaguchi. Randomized manipulation planning for a multi-fingered hand by switching contact modes. In *IEEE Int. Conf. on Robotics and Automation*, pages 2689–2694, 2003.

Chapter V

Mobile Robots Design, Modelling and Control

Angular Momentum Based Controller for Balancing an Inverted Double Pendulum

Morteza Azad* and Roy Featherstone*

* School of Engineering, Australian National University, Canberra, Australia

Abstract. This paper presents a new control algorithm, based on angular momentum, for balancing a planar inverted double pendulum robot having one degree of underactuation. The robot may either pivot about a fixed point, or roll with a curved foot over a flat ground. The controller is able to stabilize the robot in any unstable balanced configuration, and to follow arbitrary motion trajectories without losing balance. The latter necessarily involves some tracking error. Several simulation results are presented.

1 Introduction

This paper considers the problem of balancing a planar inverted double pendulum (IDP) consisting of two bodies connected by an actuated revolute joint. The lower body contains a curve, called the foot, which makes a rolling contact with a flat supporting surface (the ground). If the curve shrinks to a single point then the rolling contact simplifies to a passive revolute joint. In this case, the IDP resembles the acrobot (for acrobatic robot) which was introduced by Hauser and Murray (1990). The control problem for the acrobot has been studied by several researchers. The pioneering work of Spong (1995) used a LQR controller for balancing the acrobot in its upright equilibrium position. Xin and Kaneda (2001), Inoue et al. (2007) and Lai et al. (2005) also used the same controller for balancing the acrobot.

Berkemeier and Fearing (1999) introduced a controller based on zero dynamics for trajectory tracking of the acrobot. They found a class of interesting feasible trajectories for the acrobot and achieved theoretically accurate trajectory tracking performance using their proposed controller.

Yamakita et al. (2002), and Yonemura and Yamakita (2004) proposed an output zeroing controller for balancing the acrobot. Their proposed controller uses an output function which is defined by the angular momentum and one other new state. To work out the value of the torque, they had to calculate the third derivative of the angular momentum.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_31,

Grizzle et al. (2005) considered the general case and designed a nonlinear controller for mechanical systems with one degree of underactuation. Their output function becomes the same as that proposed by Yamakita et al. (2002) and Yonemura and Yamakita (2004) for the acrobot case. This approach also needs to work out the third derivative of the angular momentum.

In this paper we introduce a new simple controller for an IDP mechanism which is based on its angular momentum about the contact point between the robot and the ground. The new controller is able to follow setpoint commands, in which only the target configuration is given, and also motion trajectory commands, in which the motion of the actuated joint is a prescribed function of time. However, the latter necessarily involves significant tracking errors for the purpose of maintaining balance. The controller works with both a point foot and a curved foot.

2 Model of an IDP

We consider an IDP consisting of two links connected by an actuated revolute joint. The lower link contains a surface called the foot, which makes contact with the ground at a single point. Three foot shapes are considered, as shown in Figure 1: a single point, a circular arc and a general convex curve. In the first case, the contact can be modelled as a revolute joint. In the other cases, it must be modelled as a rolling-contact joint. In all three cases, the joint has a single joint variable, q_1, which is an angle. We assume that the ground is flat and horizontal, and that there is no slipping or loss of contact betwen the foot and the ground.

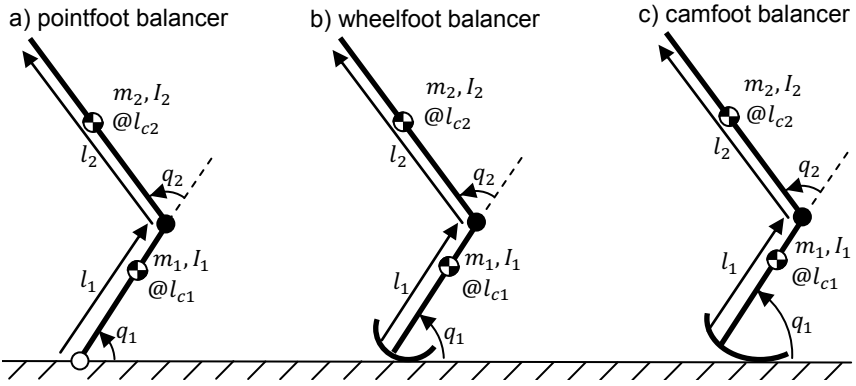


Figure 1. Robot models and parameters

3 Controlling Angular Momentum

In this section, we develop a control law for the case of a pointfoot balancer. Let X_G denote the horizontal displacement of the centre of mass (CoM) relative to the contact point, and let L denote the angular momentum of the robot about the contact point. The conditions for balance are: $X_G = 0$ and $\dot{q}_1 = \dot{q}_2 = 0$. However, as $\dot{X}_G$ and L are both linear functions of $\dot{q}_1$ and $\dot{q}_2$, the two velocity constraints can be replaced with $\dot{X}_G = 0$ and $L = 0$. Now, the rate of change of angular momentum of a multibody system about any fixed point equals the moment about that point of the external forces acting on the system. As we have chosen to express L at the contact point, which is the one point about which the moment of the ground reaction force is always zero, it follows that $\dot{L}$ must equal the moment of the gravitational force about the contact point. So we have

$$\dot{L} = -(m_1 + m_2)gX_G\,, \tag{1}$$

(where g is gravitational acceleration) and therefore also

$$\ddot{L} = -(m_1 + m_2)g\dot{X}_G\,. \tag{2}$$

The conditions for balance can now be written $L = \dot{L} = \ddot{L} = 0$; so we consider the angular momentum as an output function and define a linear feedback controller that drives the output function to zero exponentially. So the **control law** would be

$$\tau = k_{dd}\ddot{L} + k_d\dot{L} + k_pL + \tau^d\,, \tag{3}$$

where k_{dd}, k_d and k_p are controller gains, and τ^d is the correct holding torque at the actuated joint for the desired balanced configuration. The effect of τ^d is to make this configuration an equilibrium point of the closed-loop system. If q_1^d and q_2^d are the joint angles in the desired configuration, then

$$\tau^d = m_2 l_{c2} g\cos(q_1 + q_2)\,. \tag{4}$$

This controller is equivalent to

$$\tau = -k_v\dot{X}_G - k_xX_G + k_pL + \tau^d \tag{5}$$

where $k_v = (m_1 + m_2)gk_{dd}$ and $k_x = (m_1 + m_2)gk_d$. We use this alternative control law for balancers that make a rolling contact with the ground. Note that these control laws are not robust to modelling errors: if τ^d is not exactly the correct holding torque then the controller will not converge exactly to the desired configuration, but to a nearby one instead.

3.1 Stability Analysis and Gain Calculation

Considering the nonlinear state-space equations for an IDP as

$$\dot{x} = f(x) + g(x)\,.\,u \tag{6}$$

where $x = (q_1 - q_1^d, q_2 - q_2^d, \dot{q}_1, \dot{q}_2)$, and knowing that $u = \tau$ is a function of x, we will have $\dot{x} = h(x)$. Linearizing about $x = 0$ gives

$$\dot{x} = A\,x \tag{7}$$

where $A = \frac{\partial h}{\partial x}|_{x=0}$ is a 4×4 matrix. To check the stability of the system and calculate the controller gains, first we need to calculate the eigenvalues of matrix A, which are the roots of its characteristic equation. The characteristic equation for the pointfoot balancer has the general form:

$$\lambda^4 + (b_1 k_{dd} - b_2 k_p)\lambda^3 + (b_3 k_d - \alpha)\lambda^2 + (b_4 k_p)\lambda + a = 0 \tag{8}$$

where b_i, α and a are parameters which are dependent on matrix A and the desired configuration. This system is stable if all four eigenvalues are negative. Since a is always positive, one possible solution for Equation 8 would be

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = -p \tag{9}$$

so

$$(\lambda + p)^4 = \lambda^4 + 4p\lambda^3 + 6p^2\lambda^2 + 4p^3\lambda + p^4 = 0 \tag{10}$$

By this assumption, all poles of the closed loop system are negative ($p = \sqrt[4]{a}$) so the system is always stable and the controller gains become

$$k_p = \frac{4\,p^3}{b_4}\,, \qquad k_d = \frac{6\,p^2 + \alpha}{b_3}\,, \qquad k_{dd} = \frac{4\,p + b_2\,k_p}{b_1}\,. \tag{11}$$

4 Rolling Contact

Rolling contact involves a moving contact point. If $R(q_1)$ expresses the radius of curvature of the foot at the contact point as a function of the joint angle q_1, then the velocity of the contact point is $-R(q_1)\dot{q}_1$. For the pointfoot balancer, $R(q_1) = 0$ for all q_1; for the wheelfoot balancer, $R(q_1)$ is a positive constant; and for the camfoot balancer, we have only that $R(q_1) \geq 0$ for all q_1, which is the general case. Note that $R(q_1)$ is, in general, only a piecewise-continuous function of q_1. If $R(q_1)$ changes discontinuously at some value of q_1 then the velocity of the contact point also changes discontinuously.

To balance on a rolling contact, we use the control law in Equation 5 (with different values for the gains). As X_G is defined to be the horizontal position of the CoM relative to the contact point, it follows that $\dot{X}_G$ is the horizontal velocity of the CoM relative to the contact point; so $\dot{X}_G$ is capable of changing discontinuously. The method of calculating gains is the same as for the pointfoot balancer, although Equation 8 now contains a few small-magnitude terms that are nonlinear in the gains. We have ignored these terms when calculating the gains.

5 Following a Trajectory

To make the robot move from one balanced configuration to another, one simply changes τ^d to the correct value for the new configuration. However, to make the robot follow a prescribed trajectory is harder. We define an arbitrary motion trajectory for an IDP to be an equation that specifies q_2 as an explicit function of time. In general, following such a trajectory will be physically impossible without losing balance. Therefore, we expect the controller to follow the trajectory as nearly as possible subject to the constraint that it must maintain the robot's balance. This implies significant trajectory tracking errors.

To implement trajectory following, we use the control law in Equation 3, but replace L with $(L-L^d)$ and calculate τ^d differently. L^d is the theoretical value of L assuming that the robot happens to be perfectly balanced at the current instant and following the trajectory exactly; and τ^d is likewise the theoretical value of τ under these same assumptions.

6 Simulation Results

The parameters that we have used in our simulations are shown in Table 1. The radius of the foot of the wheelfoot balancer is 5cm. The foot of the camfoot balancer is composed of a clothoid curve segment and its mirror image, the two being joined to form a shield shape with a sharp point at the bottom. The radius of curvature of the clothoid is 0.15 immediately adjacent to the sharp point, and reduces gradually with increasing distance from the point. In an upright configuration, camfoot is pivotting on its sharp point. In a crouched configuration, it is rolling on the clothoid. In all cases only q_2 is controlled but the control signal is a function of both q_1 ans q_2.

6.1 Balancing in Upright and Crouched Configurations

Figures 2(a), 2(c) and 2(e) show the pointfoot, wheelfoot and camfoot balancers moving to an upright balanced position ($q_2 = 0$) from a crouched position ($q_2 = -\frac{\pi}{2}$). In each case, the initial velocity is zero and the initial value of q_1 is calculated for perfect balance given $q_2 = -\frac{\pi}{2}$. Figures 2(b), 2(d) and 2(f) show the corresponding crouching motions, in which each balancer starts in a balanced upright position ($q_2 = 0$) and moves to a balanced crouching position ($q_2 = -\frac{\pi}{2}$). The gains used for these six motions are shown in Table 2. In every case, the gains are computed by linearizing about the target position using the method described in section 3.1. The curves for the camfoot balancer show sharp changes at about $t = 0.5$ in Figure 2(e), and $t = 1$ in Figure 2(f). These are caused by the large step change in $R(q_1)$ as the balancer transitions between pivotting and rolling.

6.2 Trajectory Tracking

Figure 2(g) shows the pointfoot balancer following a trajectory comprising two steps (including one at $t = 0$), one ramp and a sine wave. The gains used in this simulation are computed by linearizing about the target position for step and ramp commands, and the midpoint position for the sine wave command. The graph shows significant tracking errors. However this is only to be expected, as the command is physically impossible to follow. Therefore, the control system must find some physically possible trajectory to follow instead. It is not known what fraction of the tracking error is physically necessary, and what fraction is due to the controller being less than perfect.

Table 1. Simulation parameters (SI units)

robots	m_1	m_2	l_1	l_{c1}	l_2	l_{c2}	I_1	I_2
pointfoot	7	7	0.5	0.5	0.75	0.75	0	0
wheelfoot & camfoot	0.5	1	0.3	0.15	0.4	0.2	$\frac{1}{12}m_1 l_1^2$	$\frac{1}{4}m_2 l_2^2$

Table 2. Gains

	balancing			crouching to $q_2^d = -\frac{\pi}{2}$		
robots	k_{dd}	k_d	k_p	k_{dd}	k_d	k_p
pointfoot	0.9704	10.2615	21.9843	1.698	11.4853	18.8457
wheelfoot	2.6545	16.3137	27.1036	1.8542	12.4549	21.0603
camfoot	0.625	8.4534	24.2363	2.093	13.0462	20.1408

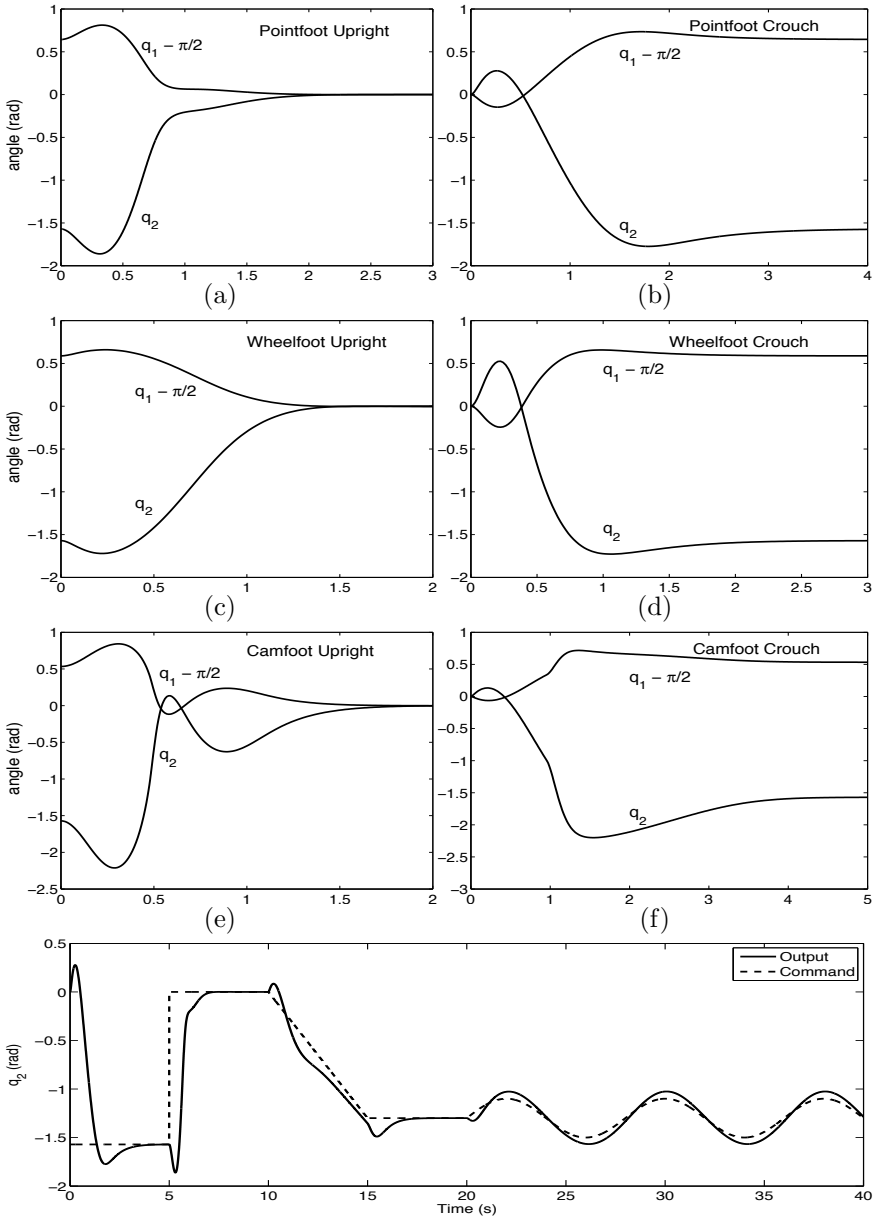


Figure 2. Simulation results for balancing upright (a, c, e), crouching (b, d, f) and trajectory tracking (g)

7 Conclusions and Future Work

In this paper we have described a new nonlinear angular momentum based balancing controller for an IDP. Simulation results show very good performance of the controller in stabilizing the IDP at unstable balanced configurations with both point and rolling contact between the robot's foot and the ground. Simulation results of tracking an arbitrary trajectory (combination of step, ramp and sine wave) for the pointfoot balancer show significant tracking errors, but some amount of tracking error is physically necessary in order to maintain balance. This work is part of a project that aims ultimately to create a machine with only two actuators that can hop and balance in 3D.

Bibliography

M. D. Berkemeier and R. S. Fearing. Tracking fast inverted trajectories of the underactuated acrobot. *IEEE Trans. Robotics and Automation*, 15(4):740–750, August 1999.

J. W. Grizzle, C. H. Moog, and C. Chevallereau. Nonlinear control of mechanical systems with an unactuated cyclic variable. *IEEE Trans. Automatic Control*, 50(5):559–576, May 2005.

J. Hauser and R. M. Murray. Nonlinear controllers for nonintegrable systems: the acrobot example. In *Proc. American Control Conf.*, pages 669–671, San Diego, CA, 23–25 May 1990.

A. Inoue, M. Deng, S. Hara, and T. Henmi. Swing-up and stabilizing control system design for an acrobot. In *Proc. IEEE Int. Conf. Networking, Sensing and Control*, pages 559–561, London, UK, 15–17 April 2007.

X. Lai, Y. Wu, J.She, and M. Wu. Control design and comprehensive stability analysis of acrobots based on lyapunov functions. *J. Central South University of Technology*, 12(1):210–216, 2005.

M. W. Spong. The swing up control problem for the acrobot. *IEEE Control Systems*, 15(1):49–55, 1995.

X. Xin and M. Kaneda. A new solution to the swing up control problem for the acrobot. In *Proc. 40th SICE Annual Conf.*, pages 124–129, Nagoya, Japan, 25–27 July 2001.

M. Yamakita, T. Yonemura, Y. Michitsuji, and Z. Luo. Stabilization of acrobot robot in upright position on a horizontal bar. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 3093–3098, Washington, DC, 11–15 May 2002.

T. Yonemura and M. Yamakita. Swing up control problem of acrobot based on switched output functions. In *Proc. SICE Annual Conf.*, pages 1909–1914, Sapporo, 4–6 August 2004.

Analysis and Design of Planar Self-Balancing Double-Pendulum Robots

Roy Featherstone*

* School of Engineering, The Australian National University, Canberra, ACT, Australia

Abstract This paper explores the attributes of a double-pendulum robot that determine its ability to balance. A new measure is defined, called the velocity gain, that expresses the degree to which the robot's centre of mass will move in response to motion of the robot's actuated joint. This measure can be used both to analyse a robot's performance and to design robot mechanisms for improved performance. Some properties of the velocity gain are explained, and several examples of both good and bad balancing robots are presented. The significance of this work is that a robot mechanism's intrinsic ability to balance sets a physical upper limit to the robot's attainable performance at balancing tasks, which is independent of the choice of control system.

1 Introduction

Balancing is an important activity for some kinds of mobile robot. A great deal of research has been devoted to the development of control systems for balancing, of which Grizzle et al. (2005); Spong (1995); Xinjilefu et al. (2009) is just a small sample; but there appears to have been no corresponding effort to study the physical attributes of a robot mechanism—its kinematic and inertia parameters—that govern its ability to perform balancing tasks.

This paper presents a study of balancing in 2D performed by a planar double-pendulum robot making a single point contact with a supporting surface, this being the simplest case of a self-balancing mobile robot. As the shape of the foot is an important factor, this paper considers the case of a general convex curve that may contain sharp points. Thus, both the case of balancing on a sharp point and balancing on a rolling contact will be considered.

To achieve and maintain balance, the control system must adjust the robot's centre of mass (CoM) indirectly via motions of the actuated joint.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_32,

The overall performance will depend on the quality of the sensors, the effectiveness of the control system, and the degree to which the CoM is affected by motions of the actuated joint. This last item is the subject of this paper: it is a property of the mechanism itself, and it sets a physical upper limit to the achievable performance of the robot.

This paper introduces a dimensionless, quantitative measure of the balancing ability of a double-pendulum robot mechanism, called the *velocity gain*. It expresses the ratio between a velocity change at the actuated joint and the resulting velocity change of the CoM. The measure is defined; a method for calculating it is described; a few basic properties are stated; and a formula is presented for the special case of a sharp pointed foot. Then the issue of parameter reduction is discussed, and how parameters can be varied without changing the velocity gain. Finally, several examples of double-pendulum mechanisms are presented, including both good and bad balancers, and balancers with unusual properties; and some general comments are made on the design of good balancers.

2 Velocity Gain

Figure 1(a) shows a planar double pendulum consisting of a leg (link 1) connected to a torso (link 2) via an actuated revolute joint (joint 2). The leg makes a single-point rolling contact with the ground, which we shall call joint 1. We assume that there is no slipping or loss of contact between the leg and the ground. Joint 1 allows a single degree of motion freedom, and can be characterized by a single joint variable, q_1, which is an angle, plus a description of the shape of the foot. (We shall assume that the ground

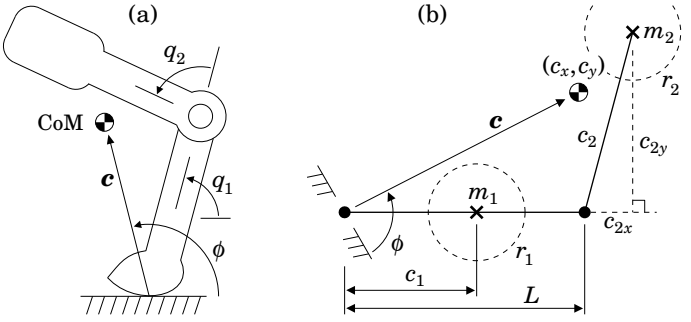


Figure 1. Definition of velocity gain (a), and parameters of a double pendulum modelled as a planar 2R mechanism (b).

is flat and horizontal.) If the contact point coincides with a sharp point on the foot, then joint 1 simplifies to a revolute joint for some range of joint angles that depends on the exact shape of the foot.

Figure 1(a) also shows a vector $\boldsymbol{c}$, which locates the robot's CoM relative to the contact point, and an angle ϕ, which gives the direction of $\boldsymbol{c}$. ϕ has been shown measured relative to the horizontal, but it could be measured relative to any desired fixed direction.

We seek a performance measure for the balancing ability of this robot. Now, the task of balancing requires control of the angle ϕ, but the only quantity that the control system can control directly is q_2. Therefore, the ability of this robot to balance depends critically on the degree to which changes in q_2 can cause changes in ϕ.

As this property will vary with configuration, let us define the following quantity as a measure of a double pendulum's balancing performance locally at configuration $\boldsymbol{q} = [q_1\ q_2]^{\mathrm{T}}$:

$$G_{\mathrm{v}}(\boldsymbol{q}) = \lim_{\Delta q_2 \to 0} \frac{\Delta\phi}{\Delta q_2} = \frac{\Delta\dot{\phi}}{\Delta\dot{q}_2}\,, \tag{1}$$

where $\Delta\dot{\phi}$ and $\Delta\dot{q}_2$ are the instantaneous changes in $\dot{\phi}$ and $\dot{q}_2$ caused by an impulsive torque applied at joint 2 with the mechanism in configuration $\boldsymbol{q}$. The magnitude of the impulse is unimportant, as it does not affect the ratio. G_{v} can also be defined as the ratio of two accelerations: $G_{\mathrm{v}}(\boldsymbol{q}) = \Delta\ddot{\phi}/\Delta\ddot{q}_2$ where $\Delta\ddot{\phi}$ and $\Delta\ddot{q}_2$ are the instantaneous changes in $\ddot{\phi}$ and $\ddot{q}_2$ due to a step change in applied torque at joint 2. Nevertheless, we shall call G_{v} the *velocity gain* of the mechanism. A double pendulum is a good balancer if $|G_{\mathrm{v}}|$ is sufficiently large at every configuration where balancing is required to take place.

Calculation Method. To calculate $G_{\mathrm{v}}(\boldsymbol{q})$, the first step is to calculate the joint-space inertia matrix at $\boldsymbol{q}$, which we shall call $\boldsymbol{H}$, and use it to work out the step response to a nonzero impulse at joint 2:

$$\begin{bmatrix} \Delta\dot{q}_1 \\ \Delta\dot{q}_2 \end{bmatrix} = \boldsymbol{H}^{-1} \begin{bmatrix} 0 \\ u \end{bmatrix}, \tag{2}$$

where $u \neq 0$ is the impulse magnitude. This equation can be solved for $\Delta\dot{q}_1$, giving

$$\Delta\dot{q}_1 = \frac{-H_{12}}{H_{11}} \Delta\dot{q}_2\,. \tag{3}$$

The next step is to calculate $\boldsymbol{c}$ and $\Delta\dot{\boldsymbol{c}}$. The latter depends on $\boldsymbol{q}$ and $\Delta\dot{q}_2$ only, as $\Delta\dot{q}_1$ can be eliminated using Eq. 3. (Bear in mind that $\dot{\boldsymbol{c}}$ gives

the relative velocity of the CoM with respect to the moving contact point.) Finally, we calculate $G_{\rm v}(\boldsymbol{q})$ from

$$G_{\rm v}(\boldsymbol{q}) = \frac{\Delta\dot{\phi}}{\Delta\dot{q}_2} = \frac{\boldsymbol{b}\cdot\Delta\dot{\boldsymbol{c}}}{|\boldsymbol{c}|\,\Delta\dot{q}_2}\,, \tag{4}$$

where $\boldsymbol{b}$ is a unit vector at right-angles to $\boldsymbol{c}$ in the direction of increasing ϕ.

A Special Case. If the robot is pivotting on a sharp point, so that joint 1 is effectively a revolute joint, then $G_{\rm v}$ depends only on q_2, and is given by the following relatively simple formula (which takes approximately one page of algebra to derive):

$$G_{\rm v}(q_2) = \frac{c_y^2 + \frac{m_2\,c_x c_{2x}}{m_1+m_2}}{c_x^2 + c_y^2} - \frac{H_{12}}{H_{11}} \tag{5}$$

where

$$\begin{aligned}
H_{11} &= m_1(r_1^2 + c_1^2) + m_2(r_2^2 + (L + c_{2x})^2 + c_{2y}^2) \\
H_{12} &= m_2(r_2^2 + c_{2x}(L + c_{2x}) + c_{2y}^2) \\
c_x &= (m_1 c_1 + m_2(L + c_{2x}))/(m_1 + m_2) \\
c_y &= m_2\,c_{2y}/(m_1 + m_2) \\
c_{2x} &= c_2\cos(q_2) \qquad \text{and} \qquad c_{2y} = c_2\sin(q_2)\,.
\end{aligned}$$

The seven parameters m_i, c_i, r_i and L define the kinematic and inertia properties of the mechanism, and are shown in Figure 1(b). r_i is the radius of gyration of link i: if I_i is the rotational inertia of link i about its CoM then $I_i = m_i r_i^2$.

Properties. $G_{\rm v}$ is defined everywhere except where $\boldsymbol{c} = \boldsymbol{0}$ and values of q_1 where there is a step change in foot curvature at the contact point. (The latter causes a step change in $G_{\rm v}$, making it only a piecewise-continuous function of q_1.) The definition of $G_{\rm v}$ as the ratio of two angular velocities makes it a dimensionless quantity. This implies that $G_{\rm v}$ is invariant with respect to a uniform scaling of the mass and/or the linear scale of the mechanism. In general, $G_{\rm v}$ depends on both q_1 and q_2, but it becomes independent of q_1 if joint 1 is revolute, and independent of q_2 if $c_2 = 0$. $G_{\rm v}$ can be positive, negative or zero, and its sign can vary with configuration. Robust balancing is impossible at or near a zero-crossing of $G_{\rm v}$.

Parameter Reduction. Whether the objective is to design or to analyse a robotic balancer, it is useful to identify a minimal set of parameters that

determine the value of G_v. The remaining parameters are then redundant in the sense that they can be varied without affecting G_v. Once these parameters are identified, one can use the minimal set to search for a mechanism with a desired property, and the redundant set to generate a family of other mechanisms with the same property.

For example, the fact that G_v is mass- and scale-invariant allows us to set the overall mass and size of the mechanism being analysed or designed to any convenient fixed value (e.g. $L = 1$ and $m_1 + m_2 = 1$ in Figure 1(b)) without loss of generality. Having found a single mechanism with an interesting property, one can immediately generate a two-parameter family of mechanisms with this same property by scaling the mass and size.

In addition to these scalings, there are potentially two more parameter reductions available. As explained in Featherstone (2008, §9.7), whenever two bodies are connected by a revolute joint, it is possible to add a particle of mass m to one of these bodies, and a particle of mass $-m$ to the other, without altering the dynamics of the mechanism. The only condition is that the two particles must coincide permanently, which implies that they must both lie on the joint's rotation axis. Applying this idea at joint 2 allows one of the remaining inertia parameters to be fixed; and it can be applied also at joint 1, if it is revolute, allowing one more parameter to be fixed. (Something similar is possible at joint 1 if the foot is a circular arc.)

To give a concrete example, if $c_2 > r_2$ in Figure 1(b) then it is possible to add a negative mass m satisfying $0 > m > -m_2$ to link 2 at joint 2 such that the modified link has $r_2 = 0$ (i.e., it has become a point mass). Repeating the process at joint 1 allows us to set $r_1 = 0$. Thus, without loss of generality, we can restrict our attention to mechanisms satisfying $r_1 = r_2 = 0$, $L = 1$ and $m_1 + m_2 = 1$, which implies that $G_\mathrm{v}(q_2)$ is in fact a function of only three independent parameters. Having found a mechanism of interest, one can immediately generate a four-parameter family of mechanisms with the same property. The zero-gain example in the next section was obtained in this way.

3 Examples

Table 1 shows the inertia and kinematic parameters of several double-pendulum mechanisms having a revolute joint 1; and Figure 2 plots their velocity gains against q_2. The parameters are as defined in Figure 1, except I_i which is defined as $I_i = m_i r_i^2$. These mechanisms are all symmetrical with respect to q_2, so it is sufficient to plot G_v in the range 0 to π.

The first example consists of two identical thin rods having unit mass, unit length and uniform mass distribution. This example has a gain of

Table 1. Parameters of several balancing mechanisms.

mechanism	m_1	c_1	I_1	m_2	c_2	I_2	L
two rods	1	0.5	0.0833	1	0.5	0.0833	1
acrobot	7	0.5	0	7	0.75	0	0.5
good balancer	0.49	0.1714	0.0036	0.11	0.4364	0.0043	0.4
zero gain	0.5	0.2	0.26	0.8	1.125	0.0675	1
zero crosser	2	0.5	0	7	0.75	0	0.5

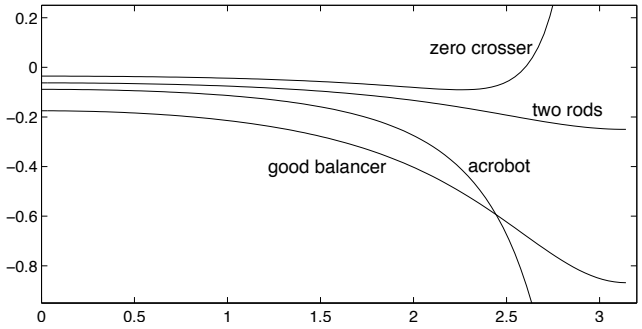


Figure 2. Velocity gain vs. q_2 for the mechanisms in Table 1.

$-1/16$ at $q_2 = 0$, increasing in magnitude to $-1/4$ at $q_2 = \pi$. That is a relatively low-magnitude gain, making this a poor balancer. Starting from an upright position, this mechanism would have to rotate q_2 by at least 16° in order to correct a 1° error in ϕ.

The next example is the acrobot as described in Berkemeier and Fearing (1998). (There is more than one published parameter set for this robot.) This mechanism is a significantly better balancer, with $G_{\mathrm{v}}(0) = -0.08867$ and $G_{\mathrm{v}}(\pi) = -3.6$. It is very common for $|G_{\mathrm{v}}|$ to be larger at π than at 0. This is partly because the CoM is closer to the foot, implying that $|\boldsymbol{c}|$ in Eq. 4 is smaller.

The next example is the result of an optimization process: starting with a mechanism consisting of a 0.4m rod weighing 0.04kg for the leg, a 0.6m rod weighing 0.06kg plus a 0.05kg point mass at the tip for the torso, and a 0.05kg point mass at joint 2 representing the joint bearing and drive, the optimization task was to find the optimal location, rounded to the nearest cm, of a 0.4kg point mass (representing an actuator) to be added to the leg. The objective function was $\int_0^{2\pi/3} G_{\mathrm{v}}(x)\,\mathrm{d}x$, and the optimal location was found to be 0.14m from the foot. This is a good balancer, having

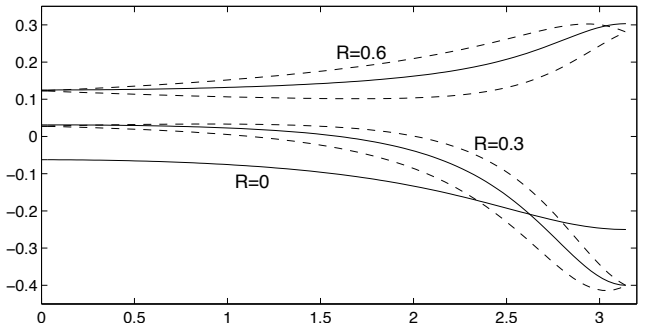


Figure 3. Effect of foot curvature on velocity gain.

$G_{\mathrm{v}}(0) = -0.175$. Generally speaking, $|G_{\mathrm{v}}(0)|$ will be large for mechanisms with a long, light torso and a short, heavy leg, the masses being concentrated at the tip of the torso and towards the bottom of the leg.

The next example has a velocity gain of zero everywhere, and is therefore impossible to balance. It was obtained by starting with the parameter set $m_1 = c_1 = 0$, $I_1 = I_2 = 0.2$ (value not critical), $m_2 = c_2 = L = 1$, which has the desired zero-gain property but specifies physically-impossible inertias, and then manipulating it in ways that do not affect G_{v} (as explained earlier) in order to find a parameter set with physically-possible inertias.

The final example shows G_{v} starting negative, but crossing zero as q_2 approaches π, ending up with $G_{\mathrm{v}}(\pi) = 5.6$. Despite the high gain at π, this is a poor balancer at $|q_2| < 2.6$. Typically, we see this pattern when joint 1 lies inside the orbit of the CoM. This example was obtained from the acrobot parameter set by reducing the mass at joint 2 until the orbit of the CoM comfortably included joint 1.

Having considered several examples where the robot is pivotting on a sharp point, so that joint 1 is effectively a revolute joint, let us now consider the case of a circular foot making rolling contact with the ground, in which case joint 1 is a rack-and-pinion joint. The value of G_{v} for a general curved foot is instantaneously identical to that for a circular foot matching the tangent and radius of curvature of the general foot at the contact point.

Figure 3 plots G_{v} against q_2 for a modified version of the two-rods example in Table 1, in which the foot is a circle of radius R centred on the point $(R, 0)$ in the coordinate system of Figure 1(b). As G_{v} now depends on q_1 as well as q_2, three curves are plotted for each value of R: one in which q_1 has been calculated to put the mechanism in its unstable balanced configuration for each given value of q_2 (solid line), and two in which q_1 is

offset by 30° each side of unstable balance (dashed lines).

Curves are plotted for $R = 0$, $R = 0.3$ and $R = 0.6$. The first is just a repeat of the two-rods curve in Figure 2, while the others show a general pattern of G_{v} becoming more positive as R increases. As a result, $R = 0.3$ makes for a very poor balancer, but $R = 0.6$ is almost twice as good as $R = 0$. For this mechanism, there is a qualitative change at $R = 0.5$, which is the point where the circle centre crosses inside the orbit of the CoM. Another qualitative change occurs at $R = 1$, which is where configuration $\boldsymbol{q} = \mathbf{0}$ switches from unstable to stable balance. Systems with $0.5 < R < 1$ may be able to exploit both stable and unstable balance, and switch between them, if the controller knows what to do.

4 Conclusion

This paper has presented a dimensionless measure, called the velocity gain, which quantifies the ability of a planar double-pendulum robot to perform balancing tasks. It can be used to analyse a given mechanism, or to design a mechanism that achieves a desired level of performance. Several examples are presented showing how the velocity gain is affected by the mechanism's kinematic and inertia parameters and the shape of the foot. The significance of this work is that velocity gain sets a physical upper limit to a robot's balancing ability, which is independent of the choice of control system. The concept of velocity gain can be extended to 3D.

Bibliography

M. D. Berkemeier and R. S. Fearing. Sliding and Hopping Gaits for the Underactuated Acrobot. *IEEE Trans. Robotics & Automation*, 14(4):629–634, 1998.

R. Featherstone. *Rigid Body Dynamics Algorithms*. Springer, New York, 2008.

J. W. Grizzle, C. H. Moog and C. Chevallereau. Nonlinear Control of Mechanical Systems With an Unactuated Cyclic Variable. *IEEE Trans. Automatic Control*, 50(5):559–576, 2005.

M. W. Spong. The Swing Up Control Problem for the Acrobot. *IEEE Control Systems Magazine*, 15(1):49–55, 1995.

Xinjilefu, V. Hayward and H. Michalska. Stabilization of the Spatial Double Inverted Pendulum Using Stochastic Programming Seen as a Model of Standing Posture Control. *Proc. 9th IEEE-RAS Int. Conf. Humanoid Robots*, Paris, France, Dec. 7–10, pp. 367–372, 2009.

Peristaltic Crawling Robot for Use on the Ground and in Plumbing Pipes

Satoshi Tesen[*] , Norihiko Saga[*] , Toshiyuki Satoh[**] and Jun-ya Nagase[***]
[*] Department of Human System Interaction, Kwansei Gakuin University, Japan
[**] Department of Machine Intelligence and Systems Engineering, Akita Prefectural University, Japan
[***] Department of Mechanical and Systems Engineering, Ryukoku University, Japan

Abstract In Japan, after the Great Hanshin Earthquake, attention became focused on rescue activities by robots on behalf of disaster victims. Each feature of the rescue robots was investigated. Attention was devoted to the peristalsis performed as a movement mechanism replaced with them when earthworm is movement. This study assesses a robot that can move forward by contact with the ground and by development of a mechanism for movement in pipes.

1 Introduction

In Japan, rescue operations performed by robots in dangerous areas and further development of robots received attention after the Hanshin Awaji earthquake disaster. Rescue operations using robots are advantageous because they can mitigate damage from secondary disasters such as fires and aftershocks that might occur in disaster areas. Particularly, the development of a robot that can search deeply into small spaces was conducted. For movement, development of the robot for use in small spaces is necessary. Movement mechanisms of various kinds such as walking, rolling on wheels, and meandering have been developed for robots. We specifically chose peristaltic crawling for three reasons. Peristaltic crawling can enable movement through narrow spaces, enable stable postures, and provide simple patterns of movement. We developed a motor-driven peristaltic crawling robot.

2 Robot Design

2.1 Expansion and contraction mechanism of the robot

The robot described herein has a mechanism to let segments expand and contract. Both segmental ends of the robot are connected in three arbors.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_33,

This axis works by operation of a motor as it expands and contracts. Figure 1 shows transport of the expansion and contraction. An aluminum Servo Horne is fixed to the body, and the robot connects it with the axis of the motor. When the motor works outward the retentive portion in the center of the segment extends outwardly as shown in Figure 1(b). The retentive portion uses friction by coming in contact with an object and fixes a body. We designate this movement as retentive. The robot realizes movement of expansion and contraction by repeating movements presented in Figure 1(a) and Figure 1(b).

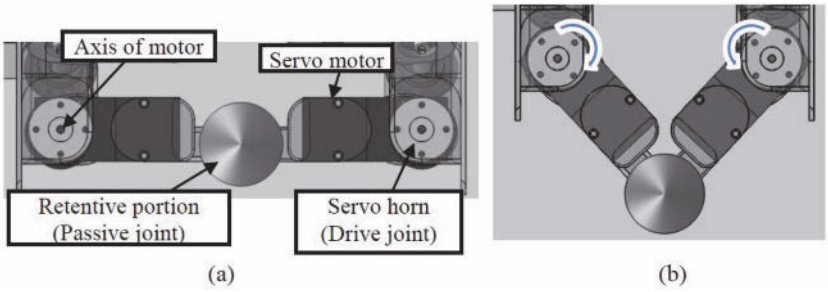


Figure 1. Mechanism of the expansion and contraction of the robot (a) expansion and (b) contraction

2.2 Requested robot specifications

We assume that the running environment of the robot is flat ground and plumbing pipes. When a robot runs on level ground, it keeps retention with two points. Moreover, when it runs in plumbing pipe, it keeps retention with three points. Diameters allowing motion in the plumbing pipe are assumed as around 250 mm and are therefore equivalent to those used for general indoor plumbing.

Next, we show the requested specifications of the part to use for the robot. The robot segments are designed by CAD and are use ABS resin. Additionally, a ball caster is attached to each segmental undersurface to reduce friction with the ground. Those features are is intended to support climbing motion in the plumbing pipe of the robot and motion by providing support for the body by maintaining retention. Furthermore, we calculated the strength of the parts and chose the form features using FEM analysis because a large load occurs in the retentive portion.

2.3 Robot climbing motion

We develop the expression of relations for retentivity necessary for the climbing motion and robot's weight. Retentivity is the power to fix segments when the robot comes into contact with an object. Therefore, if retentivity is greater than the robot's weight, then it can climb.

N (Nmm) is torque of the motor used for the robot and *L (mm)* is the length of the axis of the motor to the retentive portion. F_E is the power from the axis of the motor in L point. The power can be expressed as equation (1).

$$F_E = \frac{N}{L} \tag{1}$$

Assuming thatΘ *(degree)* is the angle of the slope of the lines which links the retentive portion to the axis of the motor, then when the state those segments shrink, the slope is 0 deg. The retentive portion moves to the wall when θ increases. $F_C(N)$ is the power with which one motor pushes the wall perpendicularly, shown by expression (2).

$$F_C = F_E \cdot \cos\theta \tag{2}$$

The power with which the one-segmental axis uses to push the wall, expressed as equation (3), where A is the number of motors used for one segmental axis.

$$F = A \cdot F_C \tag{3}$$

Additionally, we consider the frictional force with the wall called μ. The robot has 3 axis and will push 3 directions. Therefore, retentivity P to support the body becomes expression (4).

$$P = 3F \cdot \mu \tag{4}$$

The robot's weight, represented as $M(N)$, is determined while ignoring the weight of the ABS resin, considering only the motor weight. Here, $G(N)$ is the weight of one motor, *3A* is the number of the motor necessary for one segment, and T is the number of the segments. Therefore, M is expressed as expression (5).

$$M = G \cdot 3A \cdot T \tag{5}$$

Consequently, it is necessary to satisfy expression (6) to perform a climbing motion.

$$P > M \tag{6}$$

For this robot, L =90 mm, θ =60 deg, A =2, μ =0.5, and T =3. We took relations of torque N and weight G from equation (6) and perform the motor choice. Figure 2 schematizes the numerical expression described above.

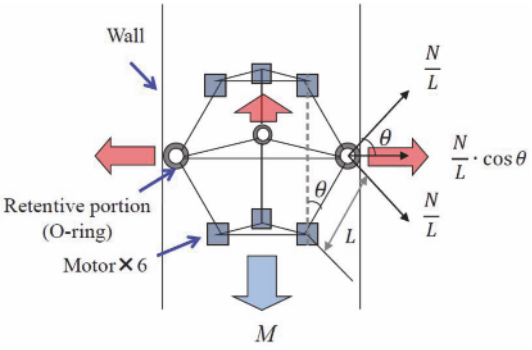


Figure 2. One-segment model of a robot for climbing motion

2.4 Strength calculation of the robot

We perform strength calculations of the part using Solid Works to inspect the transformation by maintaining retentivity. The retentive portion worked under the dynamic condition in which contact occurs many times with an object. Therefore, we chose a safety factor of 4. We used ABS Plus resin (Stratesys) for the robot body. As this reason, ABS resin is light strong and easy to mass-produce, it can exchange immediately even if a part is damaged. Material characteristics are presented in Table 1.

Table 1. Material characteristics of ABS Plus

Yield strength	28.0 MPa
Poisson ratio	0.39
Young's modulus	2000 MPa

Load conditions are such that the retentive portion receives a load from an object coming into contact perpendicularly. As a restriction condition, we fixed four places of threaded bores.

We show the result that was finally provided in Figure 3. The maximum stress has 6.6 MPa with the figure shape. It can secure a safety factor of 4. Therefore, it is thought that there is no damage in the retention movement.

2.5 Overall view of the robot

The peristaltic crawling robot which we produced to meet the requested specifications described above is depicted in Figure 4. One segment serves

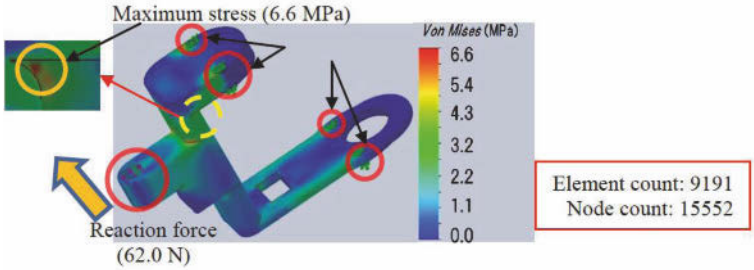


Figure 3. Result of FEM analysis.

as a symmetrical linkage mechanism, which consists of six motors. One segment of the robot can expand and contract from 116.0 mm to 183.0 mm. The maximum movement distance in a single movement is 67.0 mm. In addition, the expansion and contraction of the whole body can be 621.5 mm to 825.0 mm. The robot radius can expand and contract from 66.0 mm to 120.0 mm. The robot can run in a pipe of less than 240.0 mm and more than 132.0 mm.

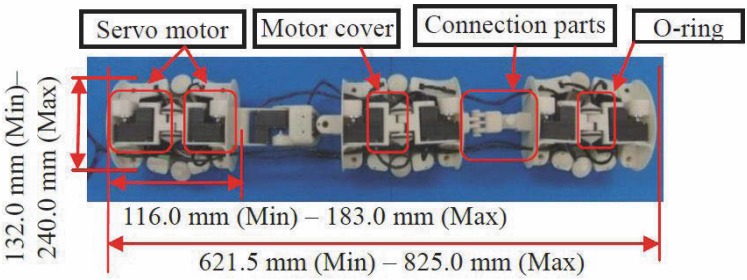


Figure 4. Robot that we developed

3 Robot design

3.1 About Q-learning

The actions treated here are not numerous. Therefore, positive convergence and action selection are performed using Q-learning, which is generally useful.

Q-learning is a commonly-used algorithm used for strengthening studies.

It is a learning algorithm that enables repetition of a trial-and-error interaction with the path in the environment. It takes an action with the highest value (Q value) of an action valuation function among possible actions in each state, as outlined in Table 2.

Table 2. Q-learning Algorithm

1. A Q value is initialized.
2. An agent observes state s_t of the environment in the time step t.
3. An agent performs action a_t in accordance with the arbitrary action selection methods.
4. Remuneration r_t is received from the environment.
5. State s_{t+1} is observed after a state change.
6. Q (s_t, a_t) is updated by the updating type of Q value, as
$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[r_t + \gamma max Q(s_{t+1},a)\right]$$
Therein, α is the learning rate ($0<\alpha\leq 1$), and γ is the discount rate ($0\leq\gamma<1$).
7. Time step t is advanced to $t+1$; it returns to 3.

3.2 Application of Q-learning

Assuming that all three segmental axes link, and that they are working, the angle can be segmental $\theta_t$on the outside, with six ways of 0 deg, 20 deg, 40 deg, 60 deg, 80 deg, or 100 deg. Regarding the robot, the combination of segment shapes becomes 6^3 ways for three segments. Furthermore, the number of actions that the robot can take next becomes 6^3. Therefore, the Q-table size is $6^3\times6^3$. The distance moved from step$t$ to $t+1$ becomes reward $r_{t.}$. The simulation is repeated for 1,000 trials. As a result of having changed a value for making trial and error, a learning rate and the value of the rate of discount produced learning rate α with 0.6 and the discount rate γ with 0.9. As a result, the movement pattern that we showed in the Figure 5 was provided. We understood that it repeated two actions. Segment 1,3 is used for the maintenance of the robot, and the segment angle change little by little from this movement. Segment 2 is used for advance, and segment angle changes for greatly.

4 Experimental results obtained for the real robot

4.1 Ground movement of the real robot

The forward motion pattern of the robot was derived by reinforcement learning. The movement pattern includes two patterns. First, the rear segment advances by forward segment retention in the body. Second, the

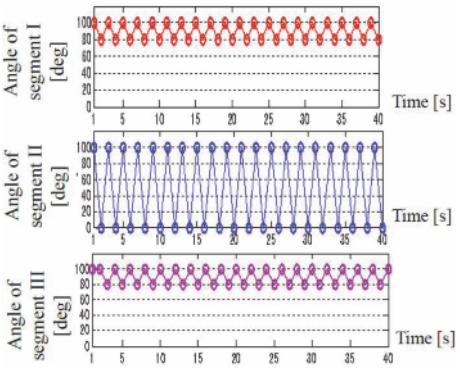


Figure 5. Optimal motion patterns of segment angles

forward segment advances by pushing of the rear segment. We show the actual movement pattern in Figure 6. 0 of Figure 6 shows the initial state. The robot repeats movements 1 to 4 and thereby performs peristaltic crawling.

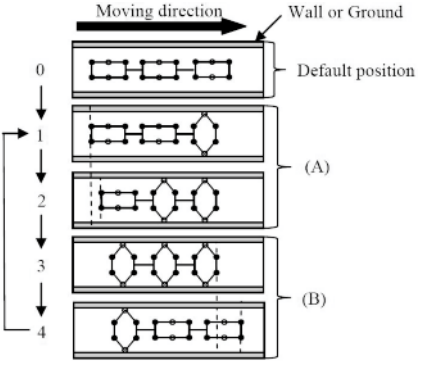


Figure 6. Moving patterns of the robot

4.2 Climbing motion of the real robot in a pipe

The robot becomes able to advance using a method resembling precession on the ground by performing movement of the retention in a pipe. We let it fit a real robot and were able to operate it, which demonstrates that the calculation of the retentivity and design are correct.

5 Conclusions

We developed a peristaltic crawling robot in this study and implemented an acquired movement pattern with simulation conducted with Q-learning and confirmed that it functioned. Furthermore, regarding the design, the robot is based on the requested specifications and exhibits high durability, retentivity, and strength as it performs calculations to determine its movements. The robot was able to function on the ground and in a pipe. Further studies will be undertaken to examine the motion for a high flexibility pattern with motor at more segmental joints of the robot.

Bibliography

R. Chatterjee, M. Nag, F. Matsuno, *Development of Modular Legged Robots:Study with Three-Legged Robot Modularity*, In *P*rocaedings of 2004 IEEE/RSJ Internalional Conference on Inlelligent Robots and Syatsma, pages 1450–1455, 2004.

H. Date, M. Sampei, S. Nakaura, *Control of a Snake Robot in Consideration of Constraint Force*, In *P*roceedings of the 2001 IEEE International Conference on Control Applications, pages 966–971, 2001

A. Menciassi, S. Gorinii G. Pemorio, P. DarioA, *SMA Actuated Artificial Earthworm*, In *P*roceedings of the 2004 IEEE International Conference on Robotics in Automation, pages 3282–3287, 2004.

K. Nagatani, D. Endo, K. Yoshida, *Improvement of the Odometry Accuracy of a Crawler Vehicle with Consideration of Slippage*, In *P*roceedings of the International Conference on Robotics and Automation, pages 2752-27579, 2007.

C. Watkins, P. Dayan, *Q-Learning, Kluwer Academic Publishers*, In *B*oston. Manufactured in The Netherlands, pages 55-68, 1992.

G. Yang, E. Chen, C. An, *Mobile robot navigation using neural Q-learning*, In *P*roceedings of the Third International Conference on Machine Learning and Cybernetics, pages 48–52, 2004

Off-road mobile robot control: an adaptive approach for accuracy and integrity

R. Lenain [1] B. Thuilot [2] N. Bouton [3] and P. Martinet [4]

[1] Irstea, 24 avenue des Landais, Clermont-Ferrand (France)
[2] Clermont Université, Université Blaise Pascal, LASMEA, Clermont-Ferrand, (France)
[3] Clermont Université, Institut Franais de Mécanique Avancée, EA 3867 - Institut Pascal UMR 6602, Clermont-Ferrand (France).
[4] Institut de Recherche en Communications et en Cybernétique de Nantes (IRCCyN), CNRS: UMR6597, Université de Nantes école Centrale de Nantes Ecole des Mines de Nantes Ecole Polytechnique de l'Université de Nantes

Abstract This paper proposes an algorithm dedicated to the control of off-road mobile robots at high speed. Based on adaptive and predictive principles, it first proposes a control law to preserve a high level of accuracy in the path tracking problem. Next, the dynamic model used for grip condition estimation is considered to address also robot integrity preservation thanks to the velocity limitation.

1 Introduction

Mobile robotics, especially in off-road context, appears as a promising answer to future needs in various fields of applications [Siegwart and Nourbakhsh (2004)], such as farming [Eaton et al. (2009)], surveillance, or military activities. In order to be efficient, such automatic devices must be accurate, efficient and robust, despite the harsh conditions encountered. If many approaches dedicated to on-road vehicles have been proposed (kinematic [Micaelli and Samson (1993)] or dynamic [Andréa-Novel et al. (1995)] model based), they are not suitable to address off-road path tracking at high speed. Indeed, they either do not describe encountered dynamic phenomena or they require the knowledge of many parameters, which can not be considered as constant in the considered context. Moreover, the fast off-road motion implies some risks for the robot integrity (rollover or spin around), which have to be accounted in the motion control.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_34,

In this paper, an adaptive and predictive approach, taking advantage of several levels of modeling is proposed. It permits to preserve the model tractability thanks to a reduced number of parameters (representative of grip conditions) on-line estimated. Based on this representation, a control law is derived for the steering angle, ensuring a good tracking accuracy whatever the grip conditions and the path to be followed, independently from the robot velocity. As a result, the robot speed can be designed in order to preserve the robot integrity, without changing the tracking performances. Then, the paper is organized as follows. First, the different levels of modeling are described and their relationships are highlighted. As this overall model requires the knowledge of some unmeasurable variables and parameters, an observer based on this multi-model point of view is developed. Once the model is entirely known, the control strategy for path tracking and integrity preservation is presented. The efficiency of the proposed approach is investigated through full-scale experiments.

2 Robot models

In order to permit an accurate off-road path tracking at high speed, the proposed control algorithm takes advantage of several levels of representation, as depicted in figure 1. The first level is based on a kinematical representation and is so called **extended kinematic model**. It allows to describe robot motion including the influence of sliding. Based on Ackermann model [Campion et al. (1993)] and detailed in [Lenain et al. (2006)], it is designed with respect to a reference trajectory, describing the evolution of the curvilinear abscissa, the lateral and angular deviations, with respect to the two control variables: the velocity at the middle of the rear axle and the front steering angle. This model accounts for sliding effects by the introduction of two sideslip angles on the two axles of the bicycle representation. The main advantage of this point of view lies in the fact that kinematic description of motion is preserved, allowing to derive a control law thanks to exact linearization techniques, as discussed in control section. Moreover, the estimation of sideslip angles at low speed can be proceeded thanks to this model (see the observer section). This level of representation is then sufficient for the motion control at limited speed. Nevertheless, dynamical effects are here neglected, and such a model appears to be inefficient when moving faster. As a consequence, a dynamic model is then required for fast sideslip angle estimation as well as for the preservation of the mobile robot integrity.

A 3D dynamical model is then considered using two 2D representations. First, the **Dynamic model 1 (yaw frame)** is still based on the bicycle

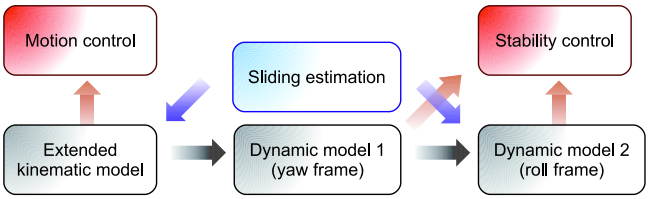


Figure 1. Synopsis of considered modeling and interaction

assumption, as achieved in [Gillespie (1992)] or in [Ben Amar and Bidaud (1995)]. In addition to variables used for the kinematic representation, the global sideslip angle and the robot inclination are introduced. Since the path tracking task is here supposed to be performed with a slow varying velocity, the longitudinal forces are neglected. As a result, only the lateral components of contact forces (for the front and rear axles) are here considered and supposed to be linearly dependent on front and rear sideslip angles. The linear coefficients (namely cornering stiffnesses) are nevertheless considered as varying (estimated by an observer), allowing to account for contact variability and tire non-linearity. The second representation, the **Dynamic model 2 (roll frame)** is fed by the outputs of dynamic variables computed thanks to the yaw frame representation. It depicts the roll motion and is devoted to the computation of the robot lateral rollover risk. Only the normal forces F_{n1} and F_{n2} (for the left and right sides) are then considered at the tire/ground contact interfaces. In order to evaluate the stability, this model is focused on the Lateral Load Transfer (LLT) computation, which is defined by $LLT = \frac{F_{n1}-F_{n2}}{F_{n1}+F_{n2}}$ and is representative of the mass repartition on robot sides. If this metric reaches ± 1, it means that two wheels of one side of the robot lift off (see [Bouton et al. (2010)] for more details).

3 Multi-model observer

As mentioned in the previous section, the proposed models can be used to proceed motion and stability controls as soon as sideslip angles and cornering stiffnesses are correctly known (i.e. with a sufficient reactivity and accuracy). The other variables and parameters can indeed be measured by the sensors on-boarded, described in section **??**, or off-line evaluated thanks to a previous calibration.

The proposed observer takes advantage of the relationship between kinematic and dynamic representations thanks to the backstepping approach depicted in figure 2. First, a **preliminary observation** based on the extended kinematic model is achieved. An estimation of sideslip angles is

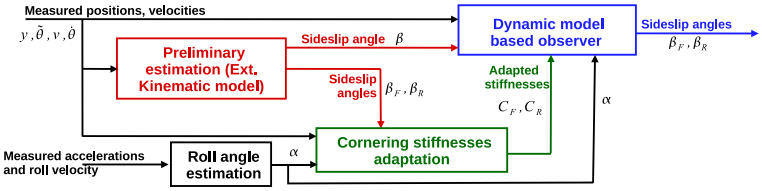


Figure 2. Global scheme of observation strategy

indeed obtained thanks to the convergence of kinematic model outputs to the measured lateral and angular deviations, as detailed in [Lenain et al. (2006)]. Alas, dynamical effects are neglected, leading to a slow-varying sideslip angle estimation, unsuitable when moving at high speed. Dynamic model 1 must be used to allow a faster adaptation. Nevertheless, such a model must be fed with the inclination of the suspended mass (measured by an accelerometer) and relevant values of cornering stiffnesses. They are on-line adapted thanks to a second step (**Cornering stiffnesses adaptation**). As slow-varying estimations of sideslip angles are available, they are used to calculate the robot global sideslip angle, considered as a measure in the sequel. A relevant value for cornering stiffnesses can then be evaluated, by ensuring the convergence of the dynamic model outputs to the measured yaw rate and global sideslip angle. The dynamic model is then totally known, and can finally be used to build an observer for the fast estimation of the sideslip angles. This last step is depicted by the box **Dynamic model-based observer** in figure 2. The detailed equations of this observer (but neglecting for the robot inclination) can be found in [Lenain et al. (2011)]. Finally, the overall dynamic model is entirely known, enabling the fast observation of sideslip angles and the control of the robot integrity.

4 Motion control and integrity preservation

4.1 Accurate trajectory control algorithm

The control law associated with the extended kinematic model is deeply detailed in [Lenain et al. (2006)], and only briefly described in this section. It is based on an exact linearization of the proposed kinematic model, specifically a conversion into a chained form. The control expression for the steering angle is then decomposed into two parts. The first term is reactive and relies mainly on current errors and observed sideslip angles. It can be then considered as adaptive, since it relies on the observed sideslip angles. The second term consists in a predictive curvature servoing using the knowledge of the reference trajectory. Based on Model Predictive Control theory,

it considers the future path curvature in order to anticipate for low level actuator delays and mobile robot inertia.

4.2 Preservation of the mobile robot integrity

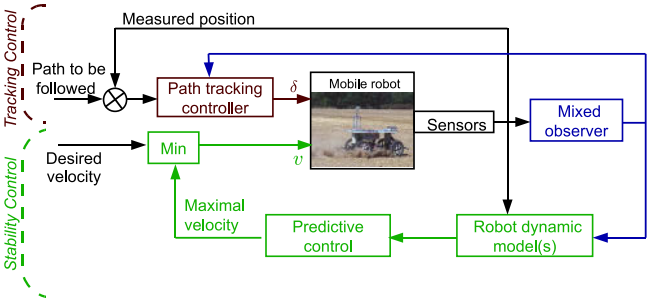


Figure 3. Global control scheme of autonomous robot

If the proposed control strategy fed with the observed sideslip angles allows to control accurately the robot motion at high speed and on different kinds of ground, it does not ensure the robot integrity. More precisely, the steering angle thus obtained, pending on the desired speed and on the terrain geometry, may generate hazardous situations, such as rollover or spin around, which are not considered within the above described path tracking control law. In order to design a robust control algorithm with respect to such phenomena, the second control variable (the robot velocity) is no more considered as a constant to be chosen at the beginning of the path tracking, but as a degree of freedom, allowing to maintain the robot in a safe behavior. Since a dynamic representation is available, the relationship between the velocity and the variables describing robot integrity can be derived. In this framework, the influence of speed on the lateral load transfer and steering angle, pending on observed grip conditions is considered. Using a predictive algorithm, the velocity leading to a chosen threshold for LLT or maximal steering angle is computed. It is then considered as the maximal velocity to be applied to the robot to preserve its integrity. The global control strategy can then be summarized by the scheme proposed in figure 3. Detailed equations for velocity limitation can be found in [Bouton et al. (2010)] for rollover prevention and in [Hach et al. (2011)] regarding the steering saturation.

5 Experimental results

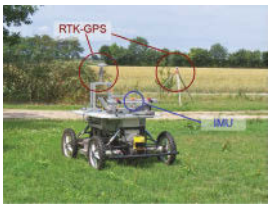
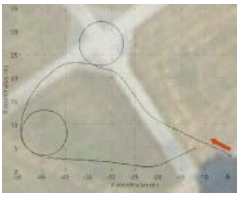
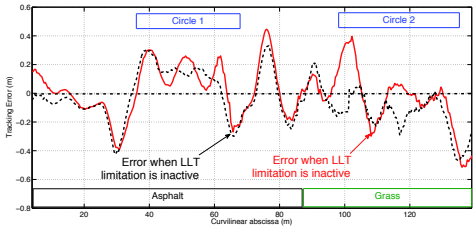


Figure 4. Experimental robot and embedded sensors

In order to study the capabilities of the proposed adaptive control strategy on uneven ground, the mobile robot depicted in figure 4 is used. This electric vehicle can reach a 8m/s velocity and is able to climb slopes up to 30°. Its weight is 450Kg and is equipped with, an RTK-GPS (giving the position of the middle of the rear axle, within an accuracy of ±2cm) and a low cost IMU (Xsens, providing three accelerations and three angular velocities). In order to point out the efficiency of the velocity moderation in order to ensure the robot integrity in the framework of path tracking, the proposed algorithm is here demonstrated only regarding the rollover prevention. The path to be followed, depicted in figure 5(a) is composed of two successive circles: one to the right, performed on asphalt, and the other to the left, performed on wet grass. The performances of the proposed algorithms are here investigated with a target speed of $4m.s^{-1}$, and path tracking is run two times: first without the velocity moderation (velocity is constant and equal to 4m/s), and secondly with the limitation active, with a desired threshold of ±0.35 for the Lateral Load Transfer.



(a) Reference path



(b) Tracking error with and without speed limitation

Figure 5. Comparison of accuracy with/without integrity preservation

We can first notice that the path tracking accuracy is independent from the velocity (both tracking errors are superposed) and despite the speed variation, depicted in figure 6, the tracking error does not exceed 40cm (during a transient phase), and does not rely on the kind of terrain (asphalt or wet grass do not influence the tracking accuracy). Classical path tracking control (typically neglecting sideslip angles), does not permit to reach such a precision, and huge errors (around 2 m, but not depicted here) are recorded during curve following on the grass part.

From a rollover point of view, when achieving the trajectory without speed moderation, the LLT reaches ± 0.4 pending on the curve and grip conditions, as depicted in figure 6 (i.e. above the desired threshold of 0.35). The interest of using speed limitation for LLT control is then clearly highlighted, since the LLT does not exceed 0.35 when integrity preservation is active.

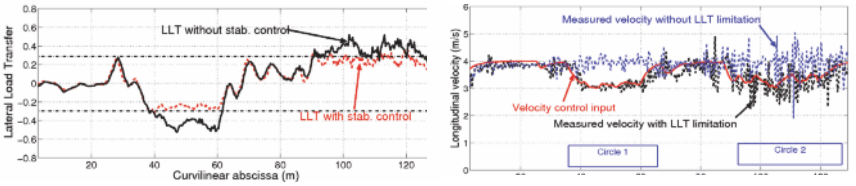


Figure 6. Speed limitation and corresponding LLT recorded (target velocity of $4m.s^{-1}$)

In order to keep the LLT within the desired range, the computed maximal speed is inferior to the desired speed of $4m.s^{-1}$ and is adapted with respect to estimated grip conditions. As it can be seen at the right in figure 6, the maximal computed speed converges to a constant value of $3m.s^{-1}$ during the first bend (on asphalt), while it varies between 2.8 up to $3.5m.s^{-1}$ during the second curve on grass, for maintaining the LLT on the desired value of 0.35. This shows the efficiency of the algorithm in preserving the robot stability and motion accuracy, whatever the ground conditions and path to be followed. Other experiments achieved at higher speed demonstrate the efficiency of the approach on both tracking accuracy and stability preservation.

6 Conclusion

This paper proposes a predictive and adaptive approach for path tracking, enabling an accurate motion control at high speed in off-road context, and ensuring the robot integrity. The efficiency of the proposed approach has been investigated through full-scale experiments. If the speed modulation permits to preserve the accuracy, the developments here proposed are part of a project[1], in which the addition of non-classical degrees of freedom (e.g active anti-roll bar) are investigated to increase the robot stability. It supposes a higher level of prediction based on Numeric Terrain Model computation.

[1]This work is supported by French National Research Agency (ANR), under the grant ANR-07-ROBO-0008 attributed to the FAST project.

Bibliography

B. d' Andréa-Novel, G. Campion, and G. Bastin. Control of wheeled mobile robots not satisfying ideal velocity constraints: a singular perturbation approach. *International Journal of Robust and Nonlinear Control*, 5(4): 243–267, 1995.

F. Ben Amar and P. Bidaud. Dynamic analysis of off-road vehicles. In *Intern. symp. on experimental robotics*, Standford, U.S.A., 1995.

N. Bouton, R. Lenain, B. Thuilot, and P. Martinet. A new device dedicated to autonomous mobile robot dynamic stability: application to an off-road mobile robot. pages 3813–3818, 2010.

G. Campion, G. Bastin, and B. d' Andréa-Novel. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. In *IEEE international conference on robotics and automation*, pages 462–469, Atlanta, Georgia (USA), 1993.

R. Eaton, H. Pota, and J. Katupitiya. Path tracking control of agricultural tractors with compensation for steering dynamics. In *International conference on decision and control (cdc)*, pages 7357–7362, Shanghai, China, 2009.

T.D. Gillespie. *Fundamentals of vehicle dynamics*. SAE International, Warrendale, U.S.A., 1992.

O. Hach, R. Lenain, B. Thuilot, and P. Martinet. Avoiding steering actuator saturation in off-road mobile robot path tracking via predictive velocity control. In *IEEE International Conference on Intelligent Robot and Systems*, volume accepted, to be published, 2011.

R. Lenain, B. Thuilot, C. Cariou, and P. Martinet. High accuracy path tracking for vehicles in presence of sliding. application to farm vehicle automatic guidance for agricultural tasks. *Autonomous robots*, 21(1): 79–97, 2006.

R. Lenain, B. Thuilot, O. Hach, and P. Martinet. High-speed mobile robot control in off-road conditions: a multi-model based adaptive approach. In *IEEE International Conference on Robotics and Automation, ICRA'11*, page 6143:6149, 2011.

A. Micaelli and C. Samson. Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. *INRIA technical report*, (n° 2097), 1993.

R. Siegwart and I.R. Nourbakhsh. *Introduction to autonomous mobile robots*. MIT Press, 2004.

On-Line Obstacle Avoidance at High Speeds

Zvi Shiller and Sanjeev Sharma
The Paslin Laboratory for Robotics and Autonomous Vehicles
Ariel University Center of Samaria, Israel

Abstract This paper presents an efficient algorithm for on-line obstacle avoidance that accounts for robot dynamics and actuator constraints. The robot trajectory (path and speed) is generated on-line by avoiding obstacles, optimally, one at a time. The trajectory is generated recursively using a *basic* algorithm that plans trajectory segments to intermediate goals. The use of intermediate goals ensures safety and convergence to the global goal. This approach reduces the original problem of avoiding m obstacles to m simpler problems of avoiding *one* obstacle each, producing a planner that is linear, instead of exponential, in the number of obstacles.

1 Introduction

This paper presents an algorithm for obstacle avoidance at high speeds. It generates trajectories that satisfy robot dynamics and actuator constraints, and are guaranteed to reach the goal from any admissible state. The incremental generation of the trajectories and the relatively low computational requirements at each step make it suitable for on-line applications.

The time-optimal control problem for robotic manipulators has been addressed in numerous studies over the past twenty years (e.g. Shiller and Dubowsky (1989)). This problem is inherently *off-line*, as it requires the solution of a two point boundary value problem. The typically nonlinear and coupled robot dynamics make such solutions computationally extensive. Adding obstacles makes the computational challenge even harder. Recent works have addressed the problem of on-line trajectory planning and high speed obstacle avoidance, using sampling-based methods (Kuwata et al., 2008; Knepper and Mason, 2009), and mixed-integer linear programming (MILP) (Vitus et al., 2008). Sampling-based methods are usually inefficient for tightly spaced environments with narrow passages. The MILP based method is computationally intensive and hence limited to very simple cases.

The on-line algorithm presented in this paper was motivated by the observation that the effect of an obstacle on the value function (the global cost-to-go function) is local. Focusing on one obstacle, while ignoring the rest,

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_35,

effectively approximates the multi-obstacle problem of avoiding m obstacles by m simpler sub-problems of avoiding one obstacle each. The trajectory is thus generated *on-line*, incrementally, one step at a time, requiring a low computational effort at each step relative to the original, inherently off-line, problem.

Convergence to the goal is guaranteed, subject to a few assumptions, by selecting at each step the obstacle with the highest cost. When those assumptions are not satisfied, the original problem is subdivided by intermediate goals to smaller problems for which the assumptions are satisfied. The algorithm is demonstrated in a numerical example for a planar point robot moving among many (70) tightly spaced circular obstacles.

2 Problem Formulation

We wish to minimize the cost function

$$\min_u \int_0^{t_f} 1 \; dt, \tag{1}$$

subject to system dynamics

$$\ddot{x} = f(x,u) \; ; x,u \in \mathbb{R}^n, \tag{2}$$

where $x \in \mathbb{R}^n$ is the vector of axis displacements, and $u \in \mathbb{R}^n$ is a vector of actuator efforts, subject to the actuator constraints, obstacle constraints and the boundary constraints:

$$|u_i| \leq 1 \; , \qquad i \in \{1, \cdots, n\} \tag{3}$$

$$g(x) \geq 0 \; ; \qquad g \in \mathbb{R}^m \tag{4}$$

$$x(0) = x_0; \; x(t_f) = x_f; \quad \dot{x}(0) = \dot{x}_0; \; \dot{x}(t_f) = \dot{x}_f \tag{5}$$

where t_f is free, and m is the number of obstacles. We assume that the obstacles (4) do not overlap with each other and with the goal x_f.

Problem (1) is difficult to solve because of the general nonlinear robot dynamics (2) and the obstacle (state) constraints (4). In addition, the obstacles introduce multiple local minima by creating multiple distinct pathways. The size of the problem can be measured by the number of such local minima. In a planar environment, each obstacle introduces two local minima (one on each side). The total number of local minima for m obstacles is therefore 2^m, which demonstrates the inherent difficulty of the time-optimal avoidance problem. This represents a computational challenge for on-line, as well as for off-line, avoidance. We settle for an approximate solution

by reducing the original problem to many simpler sub-problems that avoid obstacles optimally one at a time.

3 Optimal Avoidance of a Single Obstacle

The optimal avoidance of a single obstacle is relatively simple. We solve it for the following point mass model:

$$\ddot{x} \;=\; u_1 \;;\; |u_1| \;\leq\; 1 \qquad \ddot{y} \;=\; u_2 \;;\; |u_2| \;\leq\; 1 \tag{6}$$

where (x, y) and (u_1, u_2) represent the position and the actuator efforts, respectively, along the x and y axes. We first compute the *unconstrained* trajectory, for states not affected by the presence of the obstacle.

3.1 The Unconstrained Optimal Trajectory

The unconstrained optimal trajectory is the solution to the minimum time problem from any point in the state-space to the target state for the obstacle-free problem. For the decoupled system (6), it is determined by the minimum time of the *slowest* axis.

Consider first a single axis, represented by the double integrator

$$\dot{x}_1 = x_2 \qquad \dot{x}_2 = u; \quad |u| \leq 1 \tag{7}$$

Denoting, $x = (x_1, x_2)$, $x_0 = (x_{10}, x_{20})$ and $x_f = (x_{1f}, x_{2f})$, it can be shown that the time-optimal control for system (7) from any state x to the target x_f is bang-bang with at most one switch. The minimum *time-to-go* from any x to x_f can be computed analytically and will be skipped for the sake of brevity.

Since the minimum time trajectory for a single axis has only one switch (excluding initial states on the switching curves), reaching the target at time T greater than the minimum time, t_f, using bang-bang control, would require more than one switch (Sundar, 1994). The trajectories that reach the target state in a specified time ($T > t_f$) are not unique since the number of switches and their timing are not unique. They are, however, bounded by two bang-bang trajectories with only two switches each (Sundar, 1994).

The single axis extremal controls, $u_{max}(t)$ and $u_{min}(t)$, that generate the two extremal trajectories that reach the target at a specified time $T > t_f$, can be computed analytically:

$$u_{\max}(t) = \begin{cases} 1 & \text{if } t \in [0, t_{s1}] \\ -1 & \text{if } t \in [t_{s1}, t_{s2}] \\ 1 & \text{if } t \in [t_{s2}, T] \end{cases} \quad u_{min}(t) = \begin{cases} -1 & \text{if } t \in [0, t_{s3}] \\ 1 & \text{if } t \in [t_{s3}, t_{s4}] \\ -1 & \text{if } t \in [t_{s4}, T] \end{cases} \tag{8}$$

where

$$\begin{aligned} t_{s1} &= \frac{1}{2\alpha}(x_{1f} - x_{10} + 2\alpha T - x_{20}T - \frac{T^2}{2} - \alpha^2) \\ t_{s2} &= t_{s1} + \alpha \\ \alpha &= \frac{(T + x_{20} - x_{2f})}{2}, \end{aligned} \tag{9}$$

$$\begin{aligned} t_{s3} &= \frac{1}{2\beta}(x_{1f} - x_{10} - 2\beta T - x_{20}T + \frac{T^2}{2} + \beta^2) \\ t_{s4} &= t_{s3} + \beta \\ \beta &= \frac{(T - x_{20} + x_{2f})}{2}. \end{aligned} \tag{10}$$

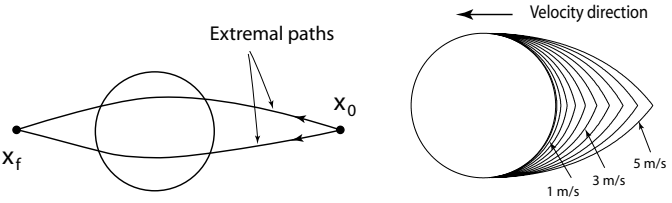
For the two axes system (6), the minimum motion time for the unconstrained problem from any state $x = (x_1, x_2, y_1, y_2)$ to the target state $x_f = (x_{1f}, x_{2f}, y_{1f}, y_{2f})$ is determined by the optimal motion time t_f of the slowest axis. The time-optimal trajectory is thus obtained by driving the slowest axis optimally, and driving the other axis so that it reaches the target at the same final time, t_f. The trajectory of the slowest axis is optimal, and unique with one switch. The trajectory of the other axis reaches the goal at a non-optimal time, and has therefore *at least* two switches, and is not unique. It follows that the time-optimal path (the projection of the trajectory to the configuration space) between the end points is not unique. The set of all time-optimal paths is bounded by two *extremal* paths, generated by the *extremal* trajectories (8) of the slowest axis.

3.2 The Constrained Optimal Trajectory

The extremal trajectories are used to determine whether an obstacle is avoidable using the *unconstrained* solution. If both extremal trajectories intersect the obstacles (see Figure 1a), then it must be avoided using the *constrained* solution discussed next. We refer to the set of points from which both extremal trajectories intersect the obstacle as the Dynamic Obstacle Shadow.

The time optimal trajectory that avoids the obstacle is computed numerically, using a line search over the terminal time. In this search, the terminal motion time is increased from the unconstrained optimal time t_f until at least one extremal avoids the obstacle. The optimal constrained trajectory has two switches in both axes.

The dynamic obstacle shadow may include *infeasible* states from which the obstacle is *unavoidable*, i.e. the entire set of attainable positions, generated by the controls satisfying (3), intersects the obstacle at some time



(a) A point in the dynamic obstacle shadow

(b) Obstacle hole

Figure 1: Obstacle shadow and obstacle hole.

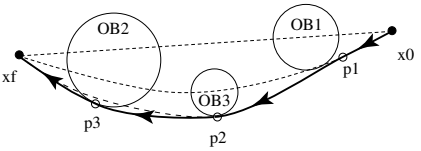


Figure 2: The avoidance procedure demonstrated for three obstacles.

$t \in [0, t_f]$. The set of all infeasible states forms, what we call, an *Obstacle Hole*, shown in Figure 1b for a circular obstacle for velocities pointing left. The *obstacle hole*, shown in Figure 1b, is represented by curves of constant speeds, generated at $0.5m/s$ increments up to $5m/s$. Each curve represents positions from which the sharpest avoidance maneuver, initiated at the corresponding velocity, is tangent to the obstacle.

4 Multi-Obstacle Avoidance

The avoidance of a single obstacle is now used to avoid multiple obstacles, *one at a time*. It remains to determine which obstacle is to be avoided at any given time. The *current* obstacle to be avoided is selected as the one that requires the longest time to avoid to the goal from the current state. This ensures that the obstacles requiring the largest deviation from the unconstrained trajectory to the goal will be considered first. This choice forms the *basic* algorithm, which under a few assumptions guarantees convergence to the goal, as discussed next.

4.1 The Basic Algorithm

The avoidance of one obstacle at a time is illustrated schematically in Figure 2 for three obstacles. The initial state, x_0, is assumed to be in the dynamic shadow of OB_1 and OB_2, hence both extremal trajectories from

that point (marked by dotted lines) cross both obstacles, as shown. Let the unconstrained optimal time from x_0 to x_f be, for example, $1s$, the constrained optimal time avoiding OB_1 be $2s$, and the constrained optimal time avoiding OB_2 be $1.5s$. OB_1 is, therefore, the maximum cost obstacle, and hence selected as the *current* obstacle. This obstacle is avoided until some point p_1, from which avoiding OB_3 takes longer than avoiding OB_1 or OB_2 to the goal. At that point, OB_3 is the maximum cost obstacle, selected to be avoided next. OB_3 is then avoided until some point p_2, from which OB_2 becomes the maximum cost obstacle. From p_3, x_f is reachable by an unconstrained optimal trajectory. This procedure is summarized in the following *basic* algorithm.

Algorithm 1: *The Basic Algorithm*

Initialize Set $x = x_0$. Select the termination condition ϵ, and time step Δt.

Step 1. Given x and x_f, determine the *current* obstacle, OB_k, to x_f. If $k = 0$, go to Step 2. Compute the optimal trajectory avoiding OB_k to x_f.

Step 2. Follow the optimal trajectory for some time step Δt. Update x. If $||x - x_f|| \leq \epsilon$, STOP.
Go to Step 1.

The basic algorithm can be proven to converge to the goal if the cost along the trajectory avoiding each obstacle decreases monotonically. This is generally the case if the following assumptions are satisfied: (i) the obstacles are convex and do not overlap; (ii) the trajectory does not enter an obstacle hole; and (iii) no two obstacles have equal cost at the same time.

The last two assumptions are generally difficult to satisfy unless the obstacles are spaced far apart. A potential violation of assumption (ii) is countered by looking one step ahead to warn of a possible fall into an obstacle hole. In the case a trajectory is about to collide with an obstacle, that obstacle becomes the *current* obstacle and is avoided next, often causing a local increase in the cost-to-go. A violation of Assumption (iii) may cause the trajectory to follow a curve that is not necessarily optimal for any obstacle, thus causing the cost along the trajectory to increase rather than decrease. This is remedied by defining an intermediate goal on the boundary of the current obstacle, and applying the *basic* algorithm to that goal. When the intermediate goal is reached, the trajectory is planned again to the global goal. This ensures that obstacles with a cost to the goal equal to that of the *current* obstacle are no longer competitive when the goal lies on the boundary of the *current* obstacle. The *basic* algorithm is thus applied recursively to each intermediate goal, until reaching the global goal.

4.2 Convergence

Convergence can be proven if assumptions (1-3) are satisfied, since the cost along the trajectory reduces monotonically by virtue of the properties of the value function. In case assumption (3) is violated, each intermediate goal subdivides the trajectory from the current state to the goal into two segments. Since every division does not involve previously avoided obstacles, the number of such divisions is finite as the number of obstacles is assumed finite. Hence, the trajectory will terminate at the final goal at a finite time after following a final number of trajectory segments.

5 Example

Figure 3 shows a trajectory of a point mass robot, computed online using Algorithm 1, that avoids 70 circular planar obstacles. It starts at the bottom right point at zero speed, and terminates at the top left corner at zero speed. The trajectory is marked by small dots, spaced at equal time intervals. The intermediate goals selected by the algorithm along the trajectory are shown as hollow circles. The large number of intermediate goals (14) reflects the tight spaces through which the robot has to maneuver. The travel time for this trajectory is $39.8s$, and its travel distance is $89.0m$ (compared to the straight line distance of $81.4m$), resulting in an average speed of $2.2m/s$. The average computation time per-step was $8ms$, for a total computation time of $3.8s$, with $\Delta T = 0.1s$, implemented in MATLAB. This example demonstrates the algorithm's successful handling of narrow passages. The algorithm was tested for 500 cases, with end points selected randomly in the same environment as in Figure 3. All 500 cases succeeded reaching the goal, demonstrating the robustness of this algorithm for a challenging environment.

6 Conclusions

An efficient approach for on-line sub-time optimal obstacle avoidance has been presented. It considers robot dynamics while attempting to minimize motion time. The algorithm is based on avoiding obstacles optimally one at a time, thus reducing the complex problem of time-optimal avoidance of m obstacles to m simpler problems, each avoiding optimally a single obstacle. This reduces the complexity of the original problem from exponential to linear in the number of obstacles. This significant reduction and the incremental nature of this approach make it suitable for on-line applications, such as mobile robots moving amongst many obstacles and in changing environments. Despite its simplicity, this approach converges to the goal from any feasible state, provided that the obstacles are avoided in the order of their

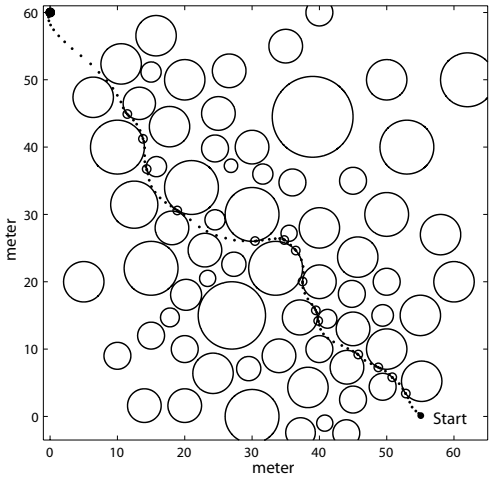


Figure 3: Navigation through 70 obstacles

cost. The algorithm was demonstrated to produce high speed trajectories through very crowded and tight spaces.

Bibliography

R.A. Knepper and M.T. Mason. Path diversity is only part of the problem. In *ICRA*, 2009.

Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How. Motion planning for urban driving using RRT. In *ICRA*, 2008.

Z. Shiller and S. Dubowsky. Time-optimal path-planning for robotic manipulators with obstacles, actuator, gripper and payload constraints. *IJRR*, 8(6):3–18, 1989.

S. Sundar. *Near-Optimal Feedback Control of Robotic Manipulators*. Ph.D. Dissertation, Dept. of Mechanical, Aerospace and Nuclear Engineering, University of California, Los Angeles, CA, 1994.

M. P. Vitus, V. Pradeep, G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin. Tunnel-milp: Path planning with sequential convex polytopes. In *AIAA Guidance, Navigation and Control Conf.*, 2008.

Torque control of a poly-articulated mobile robot during obstacle clearance

Pierre Jarrault, Christophe Grand and Phillipe Bidaud
ISIR - UPMC/CNRS (UMR 7222), Paris, France

Abstract This paper describes a control algorithm that optimises the distribution of joint torques of a polyarticulted robot while performing obstacle clearance of a large step. In this work, a special class of polyarticulted locomotion systems known as hybrid wheel-legged robots is considered. This type of system is usually redundantly actuated, involving internal forces that could be exploited to improve the tipover stability and the traction forces needed to address more challenging obstacles. The proposed algorithm is based on the forces distribution model including internal forces. Its objective is to optimize a criterion representing the maximum allowable disturbance while respecting the frictional contact conditions. The performance of this controller is evaluated in simulation.

1 Introduction

In this paper, we address the motion control of poly-articulated mobile robots during obstacle clearance. We are focusing on a special class of mobile systems, often called hybrid wheel-legged robots, which are designed in order to increase both obstacle crossing and terrain adaptation capabilities.

Systems like HyLoS (Grand et al. (2002)) and the Workpartner (Halme et al. (2003)) are examples of such hybrid locomotion systems. They are composed of 4 wheel-legs, each wheel-leg being a multi-dof serial chain ended by a driven and steerable wheel. Such robots have the ability to change the position of their center of mass (CoM) and to modify the distribution of their contact forces. Furthermore, they are often redundantly actuated systems exhibiting internal forces that should be optimized.

The proposed motion controller is based on a torque control at joint level that addresses the combined optimization of the internal forces and the CoM position, in order to maximize the contact stability (increasing traction and avoiding tip-over).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_36,

Figure 1. Hylos prototype

First, this paper introduces a stability criterion based on the contact forces slippage. Then, we present the formulation of the optimization of forces distribution. Lastly, simulation results are presented exhibiting the clearance capabilities over a step-like obstacle which height is greater than wheel radius.

2 System modelling

2.1 Equations of forces distribution

We consider a system supported by n legs (Fig. 2). The i^{th} leg is in a frictional contact with the ground at point P_i which coordinates $\boldsymbol{p}_i$ are expressed in the frame $\Re_p = (G, \boldsymbol{x}_p, \boldsymbol{y}_p, \boldsymbol{z}_p)$, attached to the chassis of the robot and located at CoM. The contact force of the ground on each leg is denoted $\boldsymbol{f}_i = \begin{bmatrix} f_{x_i} & f_{y_i} & f_{z_i} \end{bmatrix}^t$ where f_{x_i}, f_{y_i} and f_{z_i} are the components of the force along the contact frame's axis $\Re_{c_i} = (P_i, \boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{z}_i)$, such that $\boldsymbol{z}_i$ is the contact normal and $\boldsymbol{x}_i$, $\boldsymbol{y}_i$ are the tangential directions. The equations describing the equilibrium of the system are given by:

$$\mathbf{G}\,\boldsymbol{f} = \boldsymbol{F} \tag{1}$$

where $\boldsymbol{f} = \begin{bmatrix} \boldsymbol{f}_1{}^t & \dots & \boldsymbol{f}_n{}^t \end{bmatrix}^t$ is a $[3n \times 1]$ vector containing all contact forces and $\boldsymbol{F}$ is the set of external and inertial wrench applied on the platform. $\mathbf{G}$ is a $[6 \times 3n]$ matrix giving the equivalent wrench to the contact forces at the center of mass in the frame $\Re_p$ (see Grand et al. (2010) for more details).

The contact forces must respect constraints related to actuator saturation and Coulomb friction law.

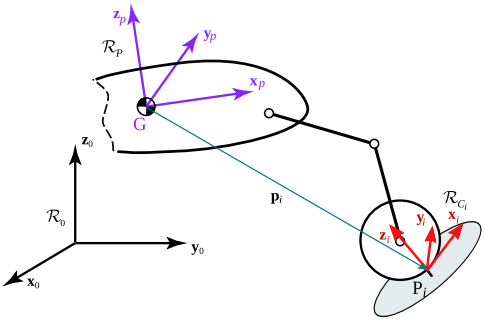


Figure 2. Model of the system

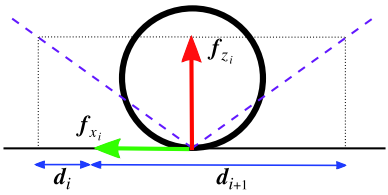


Figure 3. Residual force on a contact

The actuators limits are defined as follow:

$$\left\{ \begin{array}{rcl} \mathbf{J}^T \boldsymbol{f} & < & \boldsymbol{\tau}_{max} \\ -\mathbf{J}^T \boldsymbol{f} & < & \boldsymbol{\tau}_{max} \end{array} \right. \tag{2}$$

where $\mathbf{J} = blockdiag(\mathbf{J}_i)$, $\mathbf{J}_i$ being the Jacobian matrix of the i^{th} leg and $\boldsymbol{\tau}_{max}$ is the torque limit vector.

The contact constraints defined by Coulomb's friction law which are expressed using a pyramidal form Kerr and Roth (1986) given by:

$$\mathbf{A}_i \boldsymbol{f}_i < 0 \tag{3}$$

where

$$\mathbf{A}_i = \begin{bmatrix} 0 & 1 & \frac{\mu_i}{\sqrt{2}} \\ 0 & -1 & \frac{\mu_i}{\sqrt{2}} \\ 1 & 0 & \frac{\mu_i}{\sqrt{2}} \\ -1 & 0 & \frac{\mu_i}{\sqrt{2}} \end{bmatrix}$$

2.2 Stability criterion

Previous works on legged robot stability mostly use geometrical stability criteria based on the CoM or ZMP projection on the support polygon formed by the contact points. These methods do not consider the friction in contacts and thus are only suited for tip-over avoidance.

Extensive work has been done on the problem of forces distribution with frictional contacts in grasping researches. They express the combination of the constraints in the wrenches space associated with the CoM of the grasped object. As a result, this gives all the wrenches that are resistible if

they are applied on the CoM. Different criteria derive from this representation.

Kirkpatrick et al. (1992) uses the smallest wrench applicable on the CoM as a criterion to choose a grasp configuration. Whereas Yoshikawa and Nagai (1988) solves the forces distribution problem by maximizing the residual ball radius in this space, making sure that the robot will be able to add forces in order to support the largest possible disturbance.

However, as these methods do not consider individually each contact force, the stability of some contacts may be reduced for the sake of global stability, making modeling errors affecting the contact force control more likely to cause slippage.

Thus we propose in this paper a stability criterion based on the smallest perturbation allowed at the contact level. This leads to to maximize robustness with respect to modeling errors affecting the contact force control. Let us define vector $\boldsymbol{d}$ as:

$$\boldsymbol{d} = \mathbf{A}\ \boldsymbol{f} \tag{4}$$

where $\mathbf{A}$ is a matrix defined by $\mathbf{A} = blockdiag(\mathbf{A}_i)$.

The elements of $\boldsymbol{d}$ represent the tangential force that can be added to the contact force of a leg in each tangential direction without breaking the contact stability (Fig. 3). By maximizing the smallest element of $\boldsymbol{d}$, we maximize the perturbations that can be sustained by the robot. The optimization criterion is then:

$$\phi(\boldsymbol{f}) = \min(\mathbf{A}\ \boldsymbol{f}) \tag{5}$$

where $\min(\boldsymbol{x})$ is the function returning the smallest element of the vector $\boldsymbol{x}$. If $\phi(\boldsymbol{f})$ is negative, at least one contact is sliding or broken.

3 Optimisation of forces distribution

Considering the elements given in the previous section, the forces distribution problem is formulated as a "minimax" optimization procedure. Indeed, the objective is to maximize the smallest acceptable perturbation ($\phi = \min(\boldsymbol{d})$) subject to the constraints (1) and (2). This problem is transformed into its compact primal form (Cheng and Orin (1990)):

$$\begin{aligned} &\text{Find } \boldsymbol{x}_f \text{ maximizing:} \\ &\qquad \min\left(\mathbf{A}\left[\ \mathbf{G}^{+}\boldsymbol{F} + \mathbf{N}_f \boldsymbol{x}_f\ \right]\right) \\ &\text{Subject to:} \\ &\qquad \left[\ \mathbf{N}_f^T\ \mathbf{J} \quad -\mathbf{N}_f^T\ \mathbf{J}\ \right]^T\ \boldsymbol{x}_f < \boldsymbol{a}_f \end{aligned} \tag{6}$$

where:

- $\mathbf{G}^+\boldsymbol{F}$ represents the particular solution of (1), $\mathbf{G}^+$ being the weighted pseudo-inverse of $\mathbf{G}$,
- $\mathbf{N}_f\boldsymbol{x}_f$ represents the homogeneous solution of (1), $\mathbf{N}_f$ being the matrix containing the basis vectors of the null space of $\mathbf{G}$,
- and ${\boldsymbol{a}_f}^t = \left[\begin{array}{cc} (\boldsymbol{\tau}_{max} - \mathbf{J}^T\mathbf{G}^+\boldsymbol{F})^t & (\boldsymbol{\tau}_{max} + \mathbf{J}^T\mathbf{G}^+\boldsymbol{F})^t \end{array} \right]$

We define $\boldsymbol{x}_p = [X \quad Z]^T$ as the vector containing the longitudinal and vertical coordinates of the CoM position in the global frame (considering the global frame is chosen to have the Yaw of the robot equal to zero). In this case, the null space of $\mathbf{G}$ is entirely defined by the contact geometry. Thus, for fixed contact points, the null space of $\mathbf{G}$ is not modified by a change of X or/and Z. Furthermore for a frontal crossing, *i.e.* the rotation between the ground and the contact's frame of the obstacle is around the lateral axis of the robot ($\boldsymbol{y}_p$), the particular solution can be stated as a combination of the CoM position (X, Z) and the contact angles (represented by $\boldsymbol{f}_0$) :

$$\mathbf{G}^+\boldsymbol{F} = \boldsymbol{c}_x\ X + \boldsymbol{c}_z\ Z + \boldsymbol{f}_0 \tag{7}$$

Thus, the CoM position can be added to the optimization variables:

$$\begin{array}{ll} \text{Find } \mathbf{x} \text{ maximizing:} & \\ & \min\left(\mathbf{A}\left[\ \boldsymbol{f}_0 + \mathbf{N}\boldsymbol{x}\ \right]\right) \\ \text{Subject to} & \\ & \mathbf{J}_p^T\boldsymbol{x} < \boldsymbol{a} \\ & \|\boldsymbol{p}_i\| < D_{max} \end{array} \tag{8}$$

with

$$\begin{array}{rcl} \mathbf{N} & = & \left[\begin{array}{ccc} \mathbf{N}_f & \boldsymbol{c}_x & \boldsymbol{c}_z \end{array} \right] \\ \boldsymbol{x}^t & = & \left[\begin{array}{cc} {\boldsymbol{x}_f}^t & {\boldsymbol{x}_p}^t \end{array} \right] \\ \mathbf{J}_p & = & \left[\begin{array}{cc} \mathbf{N}^T\mathbf{J} & -\mathbf{N}^T\mathbf{J} \end{array} \right] \\ \boldsymbol{a}^t & = & \left[\begin{array}{cc} (\boldsymbol{\tau}_{max} - \mathbf{J}^T\boldsymbol{f}_0)^t & (\boldsymbol{\tau}_{max} + \mathbf{J}^T\boldsymbol{f}_0)^t \end{array} \right] \end{array}$$

4 Simulation results

The control algorithm is evaluated on the HyLoS2 robot, an actively actuated polyarticulated system consisting of 4 legs supporting the chassis. Each leg is composed of 2 segments and a wheel with four actuated degrees of freedom (DoF): two for the leg movement, one for the steering and one for the traction.

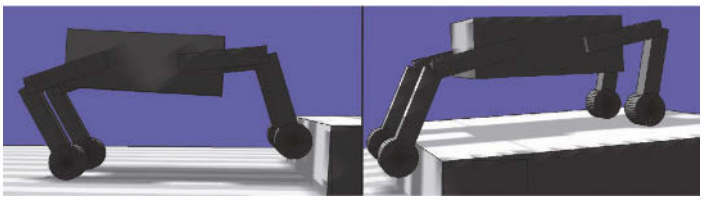


Figure 4. Simulation of the step crossing algorithm

The robot speed is sufficiently low to neglect the inertial effects on the CoM. The task of the optimisation problem (8) is only to compensate for the gravity ($\boldsymbol{F} = \begin{bmatrix} 0 & -Mg & 0 & 0 & 0 \end{bmatrix}^t$) and the displacement of the robot is created by an additional torque on the wheels. Considering this, the only CoM position's parameter which is optimized is X. During the simulation, the force optimization (6) is performed every 10 time steps of the simulation loop (1ms) while the position optimization (8) is performed every 100 steps. The null space's basis N_f is given by a singular value decomposition of the matrix G and the minimax optimization is performed by Dutta's algorithm Dutta and Vidyasagar (1977). Given a new desired configuration, the position of the CoM is controlled by adding a force to the task vector $\boldsymbol{F}$ in the force distribution problem (6). This force is computed by a PD controller. The distance between two wheels on the same side is fixed and the static friction coefficient is set to 0.8.

The robot is asked to climb a step with a height representing 25% of the distance between the wheels. The stability margin defined in (5) is plotted on Fig. 5 and the actual and desired horizontal position are given Fig. 6.

During phase 1, the robot is on a horizontal flat ground. The front wheels touch the step and it enters in phase 2, the stability margin drops below zero, indicating that it can not produce the necessary forces to start the climbing motion. The desired position is calculated and the robot reaches the optimal configuration. During this phase, the robot still uses the ground to sustain its weight. Once the stability margin has increased enough, the robot enter in phase 3 and starts to climb over the step. While climbing, the optimal position of the CoM is updated, avoiding possible tip-over as the robot is rising. Once the front wheels have reached the top of the step, the robot is once again in a phase similar to phase 1 where all the wheels are on horizontal planes. And the same phases follows: the rear wheels touch the obstacle, the robot reaches the optimal configuration and the wheels climb over the step.

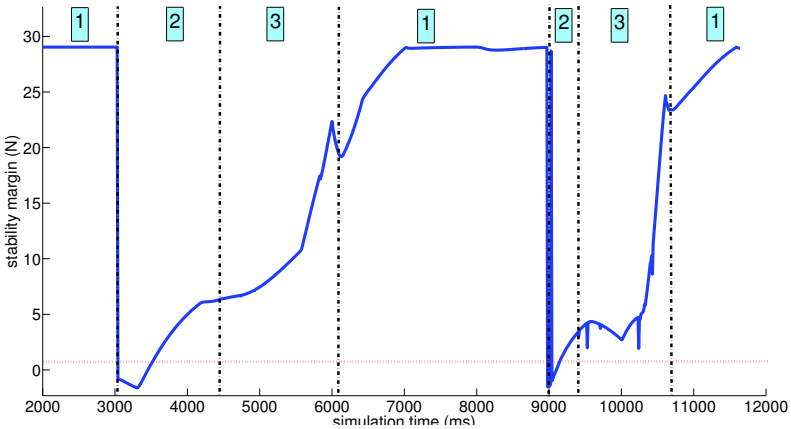


Figure 5. Stability margin during a step crossing

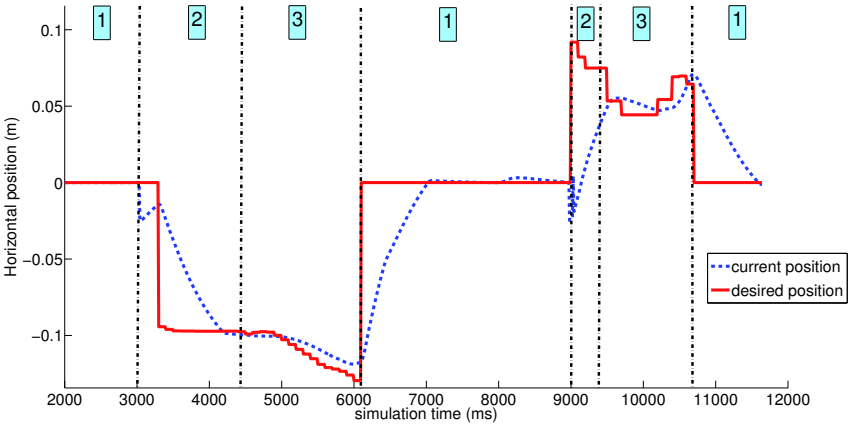


Figure 6. Desired and actual horizontal position of the CoM

5 Conclusion

The main objective of this work was to propose torque control of a poly-articulated mobile robot that exploit the redundant actuation to improve the adhesion at the wheel-ground contacts while crossing frontal step-like obstacle.

The control algorithm is based on the optimization of a criterion that represents the robustness of the contacts stability in term of traction. This criterion is based on the measure of the smallest perturbation sustainable at the contacts level. A control algorithm that compute both the best forces distribution and the best geometric configuration of the robot relative to this criterion have been described.

Simulations show good results of the control scheme for a step crossing, but an analysis of the posture trajectory shows some discontinuities that should be smoothed. Future works will address this problem by including a predictive controller and will focus on the practical implementation of this control algorithm on the actual platform HyLoS2.

Bibliography

Fan-Tien Cheng and David E. Orin. Efficient algorithm for optimal force distribution - the compact-dual lp method. In *IEEE Transaction on Robotics and Automation*, volume 6, April 1990.

S. R. K. Dutta and M. Vidyasagar. New algorithms for constrained minimax optimization. *Mathematical Programming*, 13:140–155, 1977.

C. Grand, F. Ben Amar, F. Plumet, and Ph. Bidaud. Stability control of a wheel-legged mini-rover. In *Proc. of CLAWAR'02 : 5th Int. Conf. on Climbing and Walking Robots*, pages 323–330, Paris, France, 2002.

Christophe Grand, Faiz Ben Amar, and Frederic Plumet. Motion kinematics analysis of wheeled-legged robot over 3d surface with posture adaptation. *Mechanism and Machine Theory*, 45(3):477–495, March 2010.

Aarne Halme, Ilkka Leppnen, Jussi Suomela, Sami Ylnen, and Ilkka Kettunen. Workpartner : Interactive human-like service robot for outdoor applications. *The international journal of robotics Research*, 22(7-8):627 – 640, 2003.

J. Kerr and B. Roth. Analysis of multifingered hands. *International Journal of Robotic Research*, 4:3–17, 1986.

David Kirkpatrick, Bhubaneswar Mishra, and Chee-Keng Yap. Quantitative steinitz's theorems with applications to multifingered grasping. *Discrete and Computational Geometry*, 7:295–318, 1992.

Tsunoe Yoshikawa and Kiyoshi Nagai. Evaluation and determination of grasping forces for multi-fingered hands. In *IEEE Conference on Robotic & Automation*, volume 1, pages 245–248, 1988.

An Approach to the Dynamics of a Vibration-Driven Robot

Felix Becker*, Victor Lysenko†, Vladimir Minchenya†, Igor Zeidis* and Klaus Zimmermann*

* Faculty of Mechanical Engineering, Ilmenau University of Technology, Ilmenau, Germany

† Instrumentation Engineering Faculty, Belarusian National Technical University, Minsk, Belarus

Abstract Some aspects of the motion dynamics of a body with moveable internal masses in a resisting medium have been studied. The system consists of a body with two internal masses, that can move relative to the body. One of the internal masses vibrates in the horizontal direction and the other in the vertical direction. To provide the asymmetry of the friction force, necessary for the progressive motion in a given direction, it suffices to apply bristles to the contact surfaces. The bristles are attached to the body by means of spiral springs. The equations of motion are given and solved numerically. A prototype applying these principles has been constructed and proved positive.

1 Introduction

Recent developments in mobile robotics show that, especially in medicine, biology and inspection technology, locomotion systems are needed, which can move through narrow tubular and curved environments i.e. blood vessels or pipelines. For such applications the systems should be characterised by a relative small size and weight, as well as by a simple design and the possibility of miniaturisation. Robots with a capsuled design are required for the locomotion through flowing fluids. To realise such systems the research focus has been shifted more and more towards legless (apodal) locomotion systems that are inspired by snakes, worms and similar biological objects (Zimmermann et al., 2009). Vibration-driven robots that realise the needed non-isotropic friction through evenly distributed bristles on the outer surface have a good perspective to reach the mentioned goals.

Such systems have been developed in several scientific projects. Kim et al. (2005) presented a locomotion mechanism for capsule-type endoscopes

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_37,

using shape memory alloys. Hatazaki et al. (2007) built a cilium hair based robot with only one actuator. The focus of it is to work as an active scope camera for urban search and rescue missions. In the article (Wang and Gu, 2008) a bristle-based robot for pipelines can be found. A similar idea for vibrating robotic crawlers is patented by Salomon et al. (2010). Analytical considerations of crawling robots with bristles can be found in the articles (Wang and Gu, 2008) and (Lysenko et al., 2011). Wang and Gu (2008) calculated the traction force of one bristle, based on the Euler buckling theory. Lysenko et al. (2011) modelled a single bristle as rigid body and gave the analytical solution for the velocity of the base body as well as a study of the motion behaviour of prototypes. Chernousko (2008) considered the issue of the optimal control of the motion of a body with movable internal masses.

In the present paper we consider a mathematical model of vibration-driven robots using non-isotropic friction with bristles[1]. With this multi-body system model it is possible to take into consideration the up and down movement of the main body. In a future enhancement it will be examined the rolling movement of the main body, which is present in prototypes also. The equations of motion are obtained and solved for a set of parameters numerically. A prototype and experimental investigations are given. Theory and experiments agree qualitatively.

2 Equations of Motion

The virbration-driven robot model (Figure 1) consists of a main body with the mass m_0, two internal point masses m_1 and m_2 and a number of weightless support elements, which are attached to the body by torsion springs with the stiffness c. In the case of small bending deformations such a model of rigid beams and torsion springs can be considered in a first approximation instead of an elastic body modelling. The n supports are connected to each other by a kinematic constrain so that parallelism of them remains for all times t. While the internal masses have only one motion direction each, m_1 horizontal and m_2 vertical, the body m_0 has both degrees of freedom.

In the contact points between the supports and the ground are acting dry (Coulomb) friction forces. By dry friction we understand the resistance

[1]This study was supported by the German Science Foundation (DFG) (project Zi 540/11-1).

characterised by the law

$$F = \begin{cases} k_- N, & \text{if } v < 0 \text{ or } v = 0, \text{ and } \Phi < -k_- N, \\ -k_+ N, & \text{if } v > 0 \text{ or } v = 0, \text{ and } \Phi > k_+ - N, \\ -\Phi, & \text{if } v = 0 \text{ and } -k_- N < \Phi < k_+ N, \end{cases} \tag{1}$$

where v is the velocity, N is the normal force, k_- and k_+ are the positive coefficients of friction resisting the leftward ($v < 0$) and rightward ($v > 0$) motion, respectively, and Φ is the resultant of the forces, other than frictional ones. This friction law is sometimes called Coulomb's anisotropic friction. The classical Coulomb law is a particular case of the expressions (1) for $k_- = k_+$.

Two right-handed Cartesian reference frames are introduced in the vertical plane - the fixed (initial) reference frame Oxy and the body-attached frame $O'\xi\eta$. The x- and ξ-axes are horizontal and the y- and η-axes are directed vertically upward (see Figure 1). Let x denote the coordinate of the point O' in the reference frame Oxy (the displacement of the body center of mass relative to the fixed reference frame), ξ_i and η_i the coordinates of the internal masses m_1 and m_2, respectively in the reference frame $O'\xi\eta$, and g is the gravitational acceleration. We use the Lagrange equations of

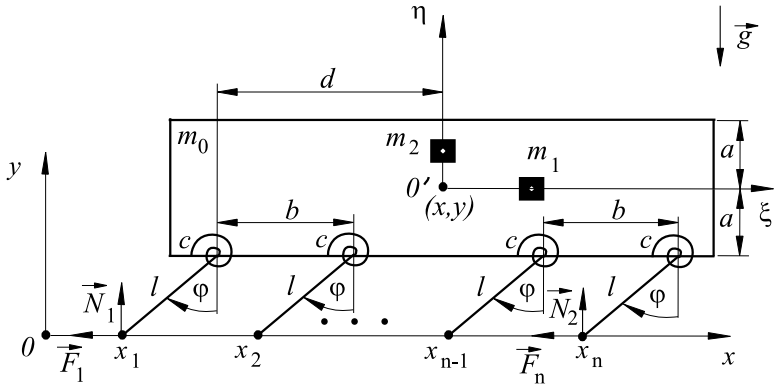


Figure 1. The schematic of the vibro-robot.

the second kind to derive the equations of motion for the system

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} = Q_i, \qquad i = 1, 2, \qquad q_1 = x, \qquad q_2 = \varphi, \tag{2}$$

where q_i are the Lagrange coordinates, T is the kinetic energy of the system and Q_i are generalized forces.

The kinetic energy of the system can be expressed as

$$T = \frac{m_0}{2}(\dot{x}^2 + \dot{y}^2) + \frac{m_1}{2}(\dot{x} + \dot{\xi})^2 + \frac{m_2}{2}(\dot{y} + \dot{\eta})^2. \tag{3}$$

The generalised forces Q_i may be presented as

$$Q_i = -\frac{\partial V}{\partial q_i} + X_i, \qquad i = 1, 2. \tag{4}$$

Here V is the potential energy of the system and X_i are non-potential generalised forces.

The potential energy of the system can be written as

$$V = (m_0 + m_1)gy + m_2 g(y + \eta) + n\frac{c}{2}(\varphi - \varphi_0)^2, \tag{5}$$

where φ_0 is the position angle of the spring in the unloaded state. The vertical coordinate of the center of mass of the body and the vertical velocity are

$$y = l\cos\varphi + a, \;\; \dot{y} = -l\sin\varphi\dot{\varphi}, \tag{6}$$

with the geometric parameters a and l, see Figure 1. By substituting (6) into the expressions (3) and (5) we obtain

$$\begin{aligned} T =& \frac{1}{2}(m_0 + m_1)\dot{x}^2 + \frac{1}{2}(m_0 + m_2)l^2\sin\varphi^2\dot{\varphi}^2 + \\ & m_1\dot{x}\dot{\xi} - m_2 l\dot{\eta}\sin\varphi\dot{\varphi} + \frac{m_1}{2}\dot{\xi}^2 + \frac{m_2}{2}\eta^2, \\ V =& (m_0 + m_1 + m_2)g(l\cos\varphi + a) + m_2 g\eta + nc(\varphi - \varphi_0)^2. \end{aligned} \tag{7}$$

The coordinates of the contact points $x_1, \ldots, x_n$, their possible displacements $\delta x_1, \ldots, \delta x_n$ and their velocities $\dot{x}_1, \ldots, \dot{x}_n$ can be written as

$$\begin{aligned} x_1 =& x - d - l\sin\varphi, & x_i =& x_1 + ib, \quad i = 1, \ldots, n-1 \\ \delta x_i =& \delta x - l\cos\varphi\delta\varphi, & \dot{x}_i =& \dot{x} - l\cos\varphi\dot{\varphi}, \end{aligned} \tag{8}$$

where d is the horizontal distance between the first support and the center of mass of the body. b is the distance between the supports, see Figure 1.

The virtual work $V\delta A$ of the friction forces $F_1, \ldots, F_n$ along the possible displacements $\delta x_1, \ldots, \delta x_n$ is given by

$$\delta A = F_1(\dot{x}_1)\delta x_1 + \ldots + F_n(\dot{x}_n)\delta x_n = Q_x\delta x + Q_\varphi\delta\varphi. \tag{9}$$

Substituting the expressions (8) into equation (9) and with respect to (4), (5) and (7) we obtain

$$\begin{aligned} Q_1 = Q_x &= F_1(\dot{x}_1) + \ldots + F_n(\dot{x}_n), \\ Q_2 = Q_\varphi &= (m_0 + m_1 + m_2)gl\sin\varphi - nc(\varphi - \varphi_0) \\ & \quad -l\cos\varphi\left(F_1(\dot{x}_1) + \ldots + F_n(\dot{x}_n)\right). \end{aligned} \tag{10}$$

The equations of motion of the system (2) with reference to (10) can be written as

$$\begin{aligned}
&(m_0+m_1)\ddot{x} = F_1(\dot{x}-l\cos\varphi\dot{\varphi})+\ldots+F_n(\dot{x}-l\cos\varphi\dot{\varphi})-m_1\ddot{\xi},\\
&(m_0+m_2)l^2\sin^2\varphi\,\ddot{\varphi}+(m_0+m_2)l^2\sin\varphi\cos\varphi\dot{\varphi}^2+2c(\varphi-\varphi_0)\\
&\quad -m_2l\sin\varphi\ddot{\eta}-(m_0+m_1+m_2)gl\sin\varphi\\
&\quad =-l\cos\varphi\left(F_1(\dot{x}-l\cos\varphi\dot{\varphi})+\ldots+F_n(\dot{x}-l\cos\varphi\dot{\varphi})\right).
\end{aligned} \tag{11}$$

The vertical coordinate y_c of the center of mass of the system can be obtained from the expression

$$(m_0+m_2)y_c = m_0y+m_2(y+\eta). \tag{12}$$

Using (12) the motion of the center of mass in the vertical direction can be expressed by

$$(m_0+m_2)\ddot{y_c} = N_1+\ldots+N_n-(m_0+m_2)g, \tag{13}$$

where N_i $(i=1,\ldots,n)$ are normal forces. From (12) using (6) we obtain

$$N_1+\ldots+N_n = (m_0+m_2)(g-l\sin\varphi\ddot{\varphi}-l\cos\varphi\dot{\varphi}^2)+m_2\ddot{\eta}. \tag{14}$$

The expression (1) for the dry friction force in the case $\dot{x}_i=\dot{x}-l\cos\varphi\dot{\varphi}\neq 0$ can be written as

$$F_i(\dot{x}-l\cos\varphi\dot{\varphi}) = -kN_i sgn(\dot{x}-l\cos\varphi\dot{\varphi}), \qquad i=1,\ldots,n. \tag{15}$$

Here

$$k=\begin{cases} k_-, & \text{if } \dot{x}-l\cos\varphi\dot{\varphi}<0,\\ k_+, & \text{if } \dot{x}-l\cos\varphi\dot{\varphi}>0. \end{cases} \tag{16}$$

From equations (14)-(16) it is obtained

$$\begin{aligned}
F_1+\ldots+F_n = &-k\left((m_0+m_2)(g-l\sin\varphi\ddot{\varphi}-l\cos\varphi\dot{\varphi}^2)+m_2\ddot{\eta}\right)\\
&\cdot sgn(\dot{x}-l\cos\varphi\dot{\varphi}).
\end{aligned} \tag{17}$$

Finally, using (17) the equations of motion of the system (11) can be represented as follows:

$$\begin{aligned}
&(m_0+m_1)\ddot{x}+m_1\ddot{\xi}\\
&+k\left((m_0+m_2)(g-l\sin\varphi\ddot{\varphi}-l\cos\varphi\dot{\varphi}^2)+m_2\ddot{\eta}\right)\\
&\cdot sgn(\dot{x}-l\cos\varphi\dot{\varphi})=0,\\
&(m_0+m_2)l^2\sin\varphi\left(\sin\varphi-k\cos\varphi sgn(\dot{x}-l\cos\varphi\dot{\varphi})\right)\ddot{\varphi}\\
&+(m_0+m_2)l^2\cos\varphi\left(\sin\varphi-k\cos\varphi sgn(\dot{x}-l\cos\varphi\dot{\varphi})\right)\dot{\varphi}^2\\
&-gl\left((m_0+m_1+m_2)\sin\varphi-k(m_0+m_2)\cos\varphi\right)sgn(\dot{x}-l\cos\varphi\dot{\varphi})\\
&+nc(\varphi-\varphi_0)+m_2l\left(\cos\varphi sgn(\dot{x}-l\cos\varphi\dot{\varphi})-\sin\varphi\right)\ddot{\eta}=0.
\end{aligned} \tag{18}$$

3 Prototype

An experimental setup is presented in Figure 2. The robot is designed for the locomotion in curved tubes with a diameter of $25mm$. The prototype consists of a bristle body made of flexible plastic foam, a source of vibration and a power supply (button cells). It has a mass of $m_0 = 6.7g$, a length of $58mm$ and a diameter of $26mm$. The bristles with an average length of $l = 8mm$ are evenly located along the cylindric body. The angle between them and normal to the body is $\varphi_0 = 30°$ in the load free state. For a high motion velocity it is useful to have the greatest bending deformation near the base body. Thus, the diameter of the bristles varies from $2mm$ (base body) to $4mm$ (free end). This geometry helps to increase the vibration amplitude at the endpoints of the bristles. The vibration of the base body is excited by a DC motor, which rotates an unbalanced mass of $m_2 = 0.9g$ with $\omega_\eta = 5000rpm$. A prototype with this parameters reaches a maximum speed of $12mm/s$ in a straight glass tube. Further experiments were made connected to the ability of the robot running up the tube under different angles. Some results are presented in Table 1. The maximum inclination angle for robot's locomotion is 45°. This behaviour is dependant on the non-isotropic friction between the robot and the tube. In the static state, without vibration, the coefficient of friction for the forward direction has a value of 0.5, for the reverse direction of 1.

Table 1. Speed of prototype in a glass tube of 25-mm-diameter.

Inclination angle [°]	0	5	10	15	20	25
Average locomotion speed $[mm/s]$	12	10	8	4	3	2

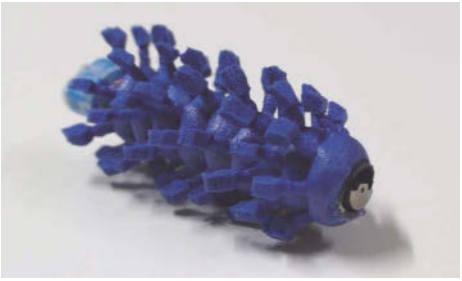


Figure 2. Prototype of the vibration-driven robot.

4 Numerical example

The numerical calculations were performed for an experimental model of the system shown in Figure 2. It is defined that $\eta(t)$ in the system of equations (18) changes harmonically $\eta(t) = a_\eta \sin(\omega_\eta t)$, where $a_\eta = 1.5mm$. The number of supports $n = 13$ and the stiffness of each torsion spring $c = 0.3Nmm$. c was obtained approximately by measuring the dependency between load and bristle deflection. The dynamic deflections of the bristles while the locomotion is less than 5°. In the case under consideration, when excitation operates only in a vertical plane, the quantity $\xi(t) = 0$. The system of equations (18) was integrated with the initial conditions $x(0) = 0,\ \dot{x}(0) = 0,\ \varphi(0) = \varphi_0,\ \dot{\varphi}(0) = 0$.

The time history of the velocity $\dot{x}(t)$ is presented in Figure 3. Figure 4 shows the time history of the angle φ. The quantity of the average velocity of the body, found on the basis of the calculations presented on Figure 3 is $13.7mm/s$. A comparison of the measured results and the numerical calculations of the average velocity of the body shows an acceptable agreement of the experiments with the theoretical predictions.

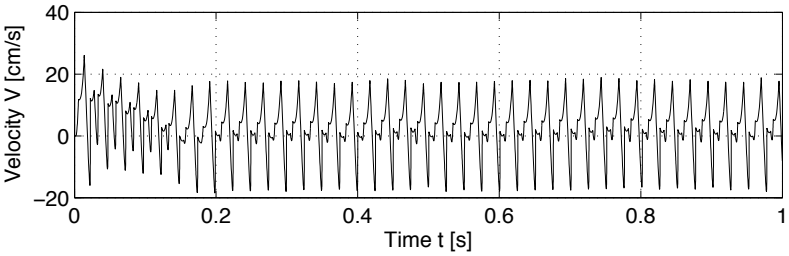


Figure 3. Velocity V over time t.

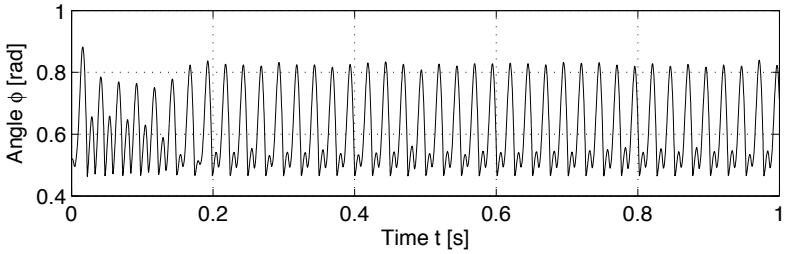


Figure 4. Angle φ over time t.

5 Conclusions

The motion of a vibration-driven robot is considered, using a multibody system model. The equations of motion are derived from the Lagrange equation of the second kind. It is shown that periodical vibrations of an internal mass in a vertical plane lead to motion of the system as a whole. The motion of the body is possible also in case of an isotropic friction. A prototype of a vibration-driven system, created on the basis of the presented concept, was designed and constructed. Future work is focused on new materials for the bristles, which should increase the efficiency of the drive system.

Bibliography

F. L. Chernousko. On the optimal motion of a body with an internal mass in a resistive medium. *J Vib Control*, 14:197–208, 2008.

K. Hatazaki, M. Konyo, K. Isaki, S. Tadokoro, and F. Takemura. Active scope camera for urban search and rescue. In *Proc. of the IEEE/RSJ Int. Conf. on Intell. Robots and Systems*, pages 2596–2602, 2007.

B. Kim, S. Lee, J. H. Park, and H.-O.Park. Design and fabrication of a locomotive mechanism for capsule-type endoscopes using shape memory alloys. *IEEE/ASME Trans. Mechatron.*, 10:77–86, 2005.

V. Lysenko, K. Zimmermann, A. Chigarev, and F. Becker. A mobile vibro-robot for locomotion through pipelines. In *Proc. of the 56th IWK, TU Ilmenau*, 2011.

O. Salomon, N. Shvalb, and M. Shoham. Vibrating robotic crawler, 04 2010. URL `http://ip.com/patapp/US20100145143`.

Z. Wang and H. Gu. A bristle-based pipeline robot for ill-constraint pipes. *IEEE/ASME Trans. Mechatron.*, 13:383–392, 2008.

K. Zimmermann, I. Zeidis, and C. Behn. *Mechanics of Terrestrial Locomotion*. Springer, 2009.

Chapter VI
Humans and Humanoids

A new approach to muscle fatigue evaluation for Push/Pull task

Ruina Ma ‡, Damien Chablat ‡ and Fouad Bennis ‡

‡ 44321 France (corresponding author: +33240376963; fax: +33240376930; e-mail: Ruina.Ma@irccyn.ec-nantes.fr).

Abstract Pushing/Pulling tasks is an important part of work in many industries. Usually, most researchers study the Push/Pull tasks by analyzing different posture conditions, force requirements, velocity factors, etc. However few studies have reported the effects of muscle fatigue. Fatigue caused by physical loading is one of the main reasons responsible for MusculoSkeletal Disorders (MSD). In this paper, muscle groups of articulation is considered and from joint level a new approach is proposed for muscle fatigue evaluation in the arms Push/Pull operations. The objective of this work is to predict the muscle fatigue situation in the Push/Pull tasks in order to reduce the probability of MSD problems for workers. A case study is presented to use this new approach for analyzing arm fatigue in Pushing/Pulling.

1 Introduction

Approximately 20% of over-exertion injuries have been associated with push and pull acts (National Institute for Occupational Safety and Health. Division of Biomedical and Behavioral Science, 1981). Nearly 8% of all back injuries and 9% of all back strains and sprains are also associated with pushing and pulling (Klein et al., 1984). Thus, there is a need to understand push and pull activities in industry since many over-exertion and fall injuries appear to be related to such activities. However, unlike other operations, push/pull capabilities have been studied only within a very limited scope. Most studies describe a laboratory experiment designed to mimic a working condition. For example: Mital did the research of push and pull isokinetic strengths from moving speed and arm angle changes (Mital et al., 1995). Badi did an experiment of one handed pushing and pulling strength at different handle heights in the vertical direction (Badi and Boushaala, 2008).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_38,

In industry a number of push/pull operations are causing the MSD problems. From the report of Health, Safety and Executive and report of Washington State Department of Labor and Industries, over 50% of workers have suffered from MSD, especially for manual handling jobs. According to the analysis in Occupational Biomechanics, muscle fatigue is an essential factor in MSD problems (Chaffin et al., 1999). Several muscle fatigue models have been proposed in the literature. In Wexler *et al.*, a muscle fatigue model is proposed based on Ca^2+ cross-bridge mechanism and verified the model with simulation experiments (Wexler et al., 1997). In Liu *et al.*, a dynamic muscle model is proposed based on motor units pattern of muscle from the biophysical point of view (Liu et al., 2002). It demonstrates the relationship among muscle activation, fatigue and recovery. Another muscle fatigue model is developed by Giat based on force-pH relationship (Giat et al., 1993). This fatigue model was obtained by curve fitting of the pH level with time t in the course of stimulation and recovery. In Ma *et al.*, a muscle fatigue model is proposed from the macroscopic point of view (Ma et al., 2009). External physical factors and personal factors are taken into consideration to construct the model from joint level. The existing muscle fatigue models consider the muscle fatigue problem from different scientific domain perspectives and each with its own advantages and disadvantages.

From the above investigation, we notice that the fatigue factor, which is actually necessary to be combined with the push/pull task, is nevertheless considered by few people. The objective of this paper is to introduce a new approach for muscle fatigue evaluation in push and pull operation and it is illustrated by using Push/Pull arm motion. Firstly, we extend the existing static muscle fatigue model in dynamic situation. Secondly, we give some hypothesis of arm muscle activities in push/pull operation. Thirdly, a new approach for muscle fatigue evaluation is proposed. Finally, a case-study is discussed to explain the two articulation push/pull operation of the arm fatigue situation in both shoulder and elbow joints.

2 A New Approach for Push/Pull Arm Fatigue Evaluation

2.1 Human muscle fatigue model

Ma proposed a muscle fatigue model from a macroscopic point of view (Ma et al., 2009). This muscle fatigue model is expressed as follows.

$$F_{\text{cem}}(t) = \text{MVC} \cdot e^{\int_0^t -k\frac{F_{\text{load}}(u)}{\text{MVC}}du} \tag{1}$$

where MVC is the maximum voluntary contraction, F_{cem} is the current exertable maximum force, $F_{\text{load}}(u)$ is the external load and k is a constant parameter. Since Ma *et al.* consider only a static situation, $F_{\text{load}}(u)$ is assumed as a constant. This muscle fatigue model was validated in the context of an industrial case study.

In dynamic situation the value of F_{load} is not a constant. According to robotic dynamic model (Khalil and Dombre, 2002), F_{load} can be modeled by a variable depending on the angle, the velocity, the acceleration and the duration of activities.

$$\mathrm{F}_{\text{load}} \stackrel{\text{def}}{=} \mathrm{F}(u, \theta, \dot{\theta}, \ddot{\theta}) \tag{2}$$

This way, Eq. (1) can be further simplified in the form.

$$\mathrm{F}_{\text{cem}}(t) = \text{MVC} \cdot e^{-\frac{k}{\text{MVC}} \int_0^t \mathrm{F}(u,\theta,\dot{\theta},\ddot{\theta})du} \tag{3}$$

Equation (3) defines the muscle fatigue model in dynamic situation. The model takes consideration of the motion by the variations of the force $\mathrm{F}_{\text{joint}}$ from joint level. This dynamic factor can be computed using Newton-Euler method or Lagrange method.

2.2 Push/Pull muscles activity hypothesis

When a human performs a simple motion, nearly all aspects in the body change. However, from a macroscopic point of view, when a human performs a push/pull operation the related muscle groups of triceps, biceps and deltoid, subscapularis are activated differently. For elbow, when a person does the push operation, the triceps muscle groups are mainly used. Inversely, when a person does the pull operation, the biceps muscle groups are mainly used. For shoulder, the movement is more complex. Generally when a person does the push operation, we can see that the deltoid related muscle groups are mainly used and inversely in the pull state the subscapularis and other related muscles are mainly used. Based on this, we suppose that every articulation is controlled by two groups muscles: *Push muscles group and Pull muscles group.* Furthermore, we suppose that during the push phase the push muscles group is in an active state and the pull muscles group is in an inactive state. The opposite would occur if the activity is inversed. Here we do not take consideration of the co-contraction factor. We can use Fig. 1(a) to illustrate the activity of push/pull muscles groups. This supposition provides the basis of the new muscle fatigue evaluation approach for push/pull tasks.

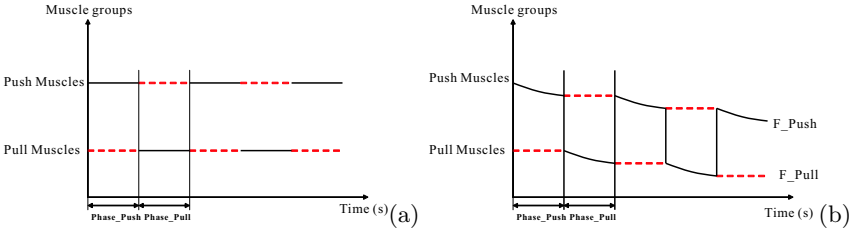


Figure 1. (a) Activity of muscle Push/Pull groups (b) Activity of muscle Push/Pull groups with fatigue

2.3 Push/Pull muscle fatigue evaluation approach

From the muscle fatigue model we can see that when a muscle is in an active state, the force generated by muscle decreases with time. In a dynamic push/pull task, the push and pull muscles groups will also experience the fatigue procedure. Based on the muscle fatigue model we introduced before, for a push muscles group, the force exerted trend with time is:

$$\mathrm{F}_{\mathrm{push\_cem}}(t) = \mathrm{MVC}_{\mathrm{push}} \cdot e^{-\frac{k}{\mathrm{MVC}_{\mathrm{push}}} \int_0^t \mathrm{F}_{\mathrm{push}}(u,\theta,\dot{\theta},\ddot{\theta})du} \tag{4}$$

Meanwhile for a pull muscles group, the force exerted trend with time is:

$$\mathrm{F}_{\mathrm{pull\_cem}}(t) = \mathrm{MVC}_{\mathrm{pull}} \cdot e^{-\frac{k}{\mathrm{MVC}_{\mathrm{pull}}} \int_0^t \mathrm{F}_{\mathrm{pull}}(u,\theta,\dot{\theta},\ddot{\theta})du} \tag{5}$$

Push/Pull task is an alternating activity. In this situation, the arm fatigue in push/pull task can be expressed by a piecewise function, as follows:

$$F_{\mathrm{push\,/\,pull}}(t) = \begin{cases} F_{\mathrm{push\_cem}}(t), & t \in \mathrm{Phase\_Push} \\ F_{\mathrm{pull\_cem}}(t), & t \in \mathrm{Phase\_Pull} \end{cases} \tag{6}$$

Figure 1(b) shows the arm fatigue situation in push/pull task.

As for the arm fatigue from a joint-based point of view, we can calculate the force of every articulation (shoulder, elbow) by using a robotic method where every external push and pull force is known. According to the Push/Pull muscle fatigue evaluation approach, when we know the external push/pull force and the arm biomechanic parameters we can know the arm fatigue trends at both the shoulder and elbow levels.

3 Case study: two articulation push/pull operation

In this case study we discuss the fatigue situation of a two articulation arm in the push/pull operation. From the joint-based view, the arm fatigue can

depend on the shoulder fatigue and elbow fatigue. Below are the detailed explanations.

3.1 Arm model

According to the biomechanical structure, we present a geometric model of the arm. We simplify the human arm to 5 revolute joints. The geometric model is shown on Fig. 2(a).

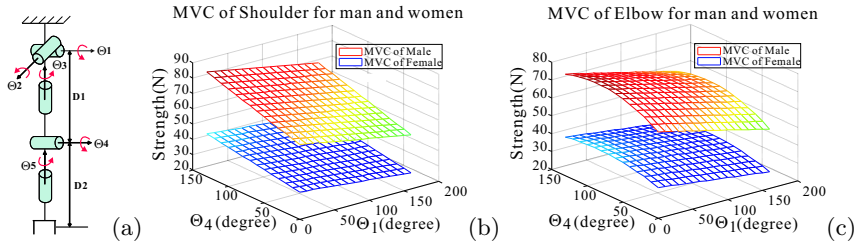


Figure 2. (a) Arm geometric model (b) Shoulder MVC of male and female (c) Elbow MVC of male and female (Chaffin et al., 1999)

3.2 Shoulder and elbow maximum strength

In the muscle fatigue equation we should pay attention especially to the MVC parameter. MVC means the maximum force that muscles generate. In static situation the MVC value is a constant, because it is decided by arm posture. Using the previously researched strength model of Chaffin (Chaffin et al., 1999), we can get that the MVC is a function dependant on the shoulder and elbow parameters. The strength model can be expressed by Eq. (7), where $\alpha_e = \theta_1$ and $\alpha_s = 180 - \theta_4$. $G_{\text{elbow}}(Male : 0.1913, Female : 0.1005)$ and $G_{\text{shoulder}}(Male : 0.1495, Female : 0.2845)$ are the parameters for gender adjustment. Fig. 2(b,c) shows the strength of elbow and shoulder in different arm angles both for man and women.

$$\begin{aligned} \text{Strength}_{\text{elbow}} &= (336.29 + 1.544\alpha_e - 0.0085\alpha_s^2) \cdot G_{\text{elbow}} \\ \text{Strength}_{\text{shoulder}} &= (227.338 + 0.525\alpha_e - 0.296\alpha_s) \cdot G_{\text{shoulder}} \end{aligned} \tag{7}$$

3.3 Fatigue risk force

When people do some tasks, at what point will they risk fatigue? A subjective measure of fatigue is not appropriate to quantify fatigue. We need an objective parameter to measure the fatigue risk. The Maximum Endurance Time (MET) is an important concept in ergonomics. It describes

the duration from the start to the instant at which the strength decreases to below the torque demand resulting from external load. Once the external load exceeds the current force capacity, potential physical risks might occur to the body tissues. So the fatigue risk force, noted $F_{\text{fatigue\_risk}}$ is reached when the external load equals the current force capacity,

$$F_{\text{fatigue\_risk}} = F_{\text{external}} = F_{\text{joint}}. \tag{8}$$

For a static posture, $F_{\text{fatigue\_risk}}$ is a constant value, because in the static situation the external force is a constant. Meanwhile, in dynamic situation, $F_{\text{fatigue\_risk}}$ is a non-linear function; it changes according to different postures. According to the arm geometric model, the force of every joint can be represented by $F_{\text{joint}} \in \{F_{\theta_1}, \ldots, F_{\theta_5}\}$. For example, $F_{\text{fatigue\_risk}}$ of shoulder is $F_{\text{shoulder}} = F_{\theta_1}$ and $F_{\text{fatigue\_risk}}$ of elbow is $F_{\text{elbow}} = F_{\theta_4}$. When the arm does the horizontal push and pull operation and the hand move alone x axis between (0.3, 0.4) meter, the shoulder angle changes between (-49.3, -42.3) degrees and the elbow angle changes between (124.1, 101.3) degrees. The force changes of shoulder and elbow were shown on Fig. 3. This also represents the fatigue risk force trends in the push/pull operations. Forces of shoulder and elbow are computed using Newton-Euler method (Khalil and Dombre, 2002).

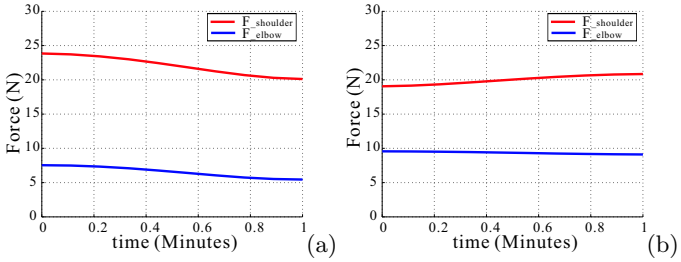


Figure 3. (a) Push Operation (b) Pull Operation

3.4 Dynamic push/pull simulation

Suppose the weight of the object is 2Kg, and a person (Sex: man, Height: 188cm, Weight: 90kg) uses 10N to push and 10N to pull this object. In reality the push/pull task time may not be exactly the same, here we simplify the situation and consider that the two operations use the same time ($T_{\text{push}} = T_{\text{pull}} = 1$ minute). Fatigue rate parameter k depends on the subject and experiment measures are needed to determine it. Here for this

specific case we choose $k = 1$ for both push and pull operations. In different postures, according to the estimated arm's strength we know the MVC of the shoulder and elbow at every moment; according to the push force and pull force we can calculate the force of elbow and shoulder at every moment. Furthermore we can know the fatigue risk line of the shoulder and elbow. Based on these initial conditions, and using our approach, we can determine the fatigue trends in shoulder and elbow separately. Fig. 4 shows the simulation of arm fatigue in both shoulder and elbow level.

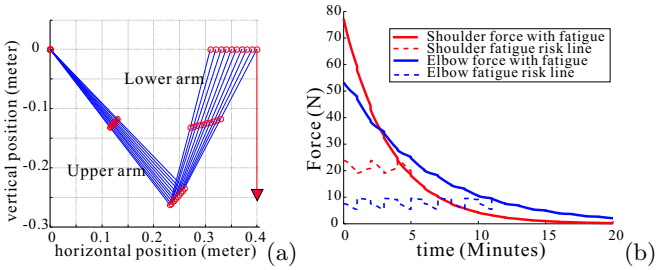


Figure 4. (a) Arm Push/Pull Operation (b) Fatigue simulation of arm Push/Pull Operation

3.5 Discussion

From the simulation result we can see that in push/pull task, the shoulder exerted force reaches $F_{\text{fatigue\_risk}}$ in about 5 minutes and the elbow exerted force reaches $F_{\text{fatigue\_risk}}$ in about 11 minutes. This explains why many workers have problems with shoulder rather than elbow from pushing/pulling. These results support the idea that when workers do this kind of work, it is better to have a rest or to change the working posture every 5 minutes to avoid fatigue. The simulation here is a general case; the specific prediction is decided by the individual person, the working posture, the weight of the object and the single push/pull operation time. In future work we will use experimental data to verify this fatigue evaluation approach. The experiment would also consider more muscles and joints, female subjects and changing external loads.

4 Conclusions

In this paper, we took into consideration the muscle groups of articulation in push and pull operations and from joint level proposed a new approach to

muscle fatigue evaluation for push/pull tasks. From simulation results we can see that in a push/pull process, the arm fatigue trends appear in both shoulder and elbow level and it is decided by the parameters MVC, F_{load}, k, and working postures. The case study shows that our new fatigue evaluation approach has a potential to provide information to help people prevent fatigue risk and estimate the safe working periods for pushing/pulling. Our work enlarges the vision of push/pull operation research. The final goal of our work is to use this model in ergonomic evaluation procedures, to enhance work efficiency and reduce MSD risks.

Bibliography

T. H. Badi and A. A. Boushaala. Effect of one-handed pushing and pulling strength at different handle heights in vertical direction. *Engineering and Technology*, page 47, 2008.

D. B. Chaffin, G. B. J. Andersson, and B. J. Martin. *Occupational Biomechanics*. John Wiley and Sons, Inc, third edition, 1999.

Y. Giat, J. Mizrahi, and M. Levy. A musculotendon model of the fatigue profiles of paralyzed quadriceps muscle under fes. *IEEE transactions on biomedical engineering*, 40(7):664–674, 1993.

W Khalil and E Dombre. *Modelling, identification and control of robots*. Hermes Science Publications, 2002.

B. P. Klein, R. C. Jensen, and L. M. Sanderson. Assessment of workers' compensation claims for back strains/sprains. *Journal of Occupational Medicine*, 26(6):443–448, 1984.

J. Z. Liu, R. W. Brown, and G.H. Yue. A dynamical model of muscle activation, fatigue, and recovery. *Biophysical journal*, 82(5):2344–2359, May 2002. ISSN 0006-3495.

L. Ma, D. Chablat, F. Bennis, and W. Zhang. A new simple dynamic muscle fatigue model and its validation. *International Journal of Industrial Ergonomics*, 39(1):211 – 220, 2009. ISSN 0169-8141.

A. Mital, P. Kopardekar, and A. Motorwala. Isokinetic pull strengths in the vertical plane: effects of speed and arm angle. *Clinic Biomechanics*, 10(2):110–112, 1995.

National Institute for Occupational Safety and Health. Division of Biomedical and Behavioral Science. *Work practices guide for manual lifting*. DHHS publication, no. (NIOSH) 81-122, 1981.

A. S. Wexler, J. Ding, and S. A. Binder-Macleod. A mathematical model that predicts skeletal muscle force. *IEEE transactions on biomedical engineering*, 44(5):337–348, 1997.

Calibration of the Human-Body inertial Parameters Using Inverse Dynamics, LS technique and anatomical values

Gentiane VENTURE* and Maxime GAUTIER**

* Tokyo University of Agriculture and Technology

** University of Nantes and the Institut de Recherche en Communication et Cybernetique de Nantes

Abstract Using the *essential parameters* of the human body, we propose to calculate the LS solution with SVD factorization, which is the closest in 2 norm of a set of a priori anatomic values given by literature database. This solution keeps both the same minimum norm error given by the *essential parameters* and the physical anatomical meaning of the a priori values when the measuring noise and errors are small. Experimental results are presented.

1 Introduction

In the past years, robotics technology and formalism have been extensively used in a variety of fields. For robots to be more realistic androids, to understand human behavior and human intentions, to reproduce skilled human tasks, fields such as learning, motion understanding, dynamics modeling are very active Khatib and et al. (2003), Mataric (1997). In particular, the human body modeling and understanding, using concepts from robotics is now used in rehabilitation, ergonomics, sport science. Human models are used for simulations, computations, design. Kinematics models are the most common. However dynamics models, to compute the interaction with the environment, generated torques are now spreading. To obtain satisfactory results, the motion measurement and the models must be accurate enough. Motion capture technology enables a precise measure. Whole body motions can be recorded within a few millimeters precision. The kinematics of the human body, due to the numerous degrees of freedom (DOF), is complex and there is no modeling norm. On the other hand, a precise dynamics calibration of the segment inertial parameters (SP) is difficult and requires abundant measurements. Several techniques to measure the segment parameters (SP) can be found in the literature: extrapolation from MRI using

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_39,

density of tissues Cheng (2000), extrapolation from geometry, mass and literature tables Dumas and et al. (2007). More recently, the use of robotics formalism has provided an in-vivo identification method, making use of motion capture and contact force measurements, for the base parameters. It has been combined with literature interpolation to obtain all the SP using constraint to guaranty physically consistency, a non-linear criteria has been added Ayusawa and Nakamura (2010). However, the obtained solution does not minimize the residual error. We propose to calibrate the identified SP with respect to a priori known consistent values using a method based on the SVD decomposition of the regressor matrix.

2 Identification Model

The inverse dynamics of a mobile system is represented by (1) Fujimoto and et al. (1998). The upper part (6 first rows) of the equation describes the 6 DOF of the root-link. The lower part (N rows) describes the motion of the N bodies of the various chains constituting the whole system.

$$\begin{bmatrix} \boldsymbol{H}_O & \boldsymbol{H}_{OC} \\ \boldsymbol{H}_{CO} & \boldsymbol{H}_C \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}}_0 \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{b}_O \\ \boldsymbol{b}_C \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\tau} \end{bmatrix} + \sum_{k=1}^{N_c} \begin{bmatrix} \boldsymbol{K}_{kO} \\ \boldsymbol{K}_{kC} \end{bmatrix} \boldsymbol{F}_k \tag{1}$$

Where $_O$ and $_C$ indexes represent respectively terms for the base link and term for the kinematic chain, $\boldsymbol{H}_{ij}$ is the inertia matrix, $\boldsymbol{q_O}$ is the vector of generalized coordinates which represents the 6 DOF of the root-link, $\boldsymbol{\theta}$ is the vector joint angles, $\boldsymbol{b}_i$ is the bias force vector including centrifugal, Coriolis and gravity forces, $\boldsymbol{\tau}$ is the vector of joint torques, N_c is the number of contact point with the environment, $\boldsymbol{F}_k$ is the k^{th} vector of external forces exerted to the robot, $\boldsymbol{K}_{ki}$ the matrices which map $\boldsymbol{F}_k$ to the generalized force vector. As shown in Khalil and Dombre (2002),(1) is linear with respect to the inertial parameters; and we obtain the *identification model* (2) by separating the vector of inertial parameters $\boldsymbol{\phi}$ from the regressor $\boldsymbol{W}$.

$$\begin{bmatrix} \boldsymbol{W}_O \\ \boldsymbol{W}_C \end{bmatrix} \boldsymbol{\phi} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\tau} \end{bmatrix} + \sum_{k=1}^{N_c} \begin{bmatrix} \boldsymbol{K}_{kO} \\ \boldsymbol{K}_{kC} \end{bmatrix} \boldsymbol{F}_k \tag{2}$$

$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_O \\ \boldsymbol{W}_C \end{bmatrix}$ is the regressor, a function of the joint angle $\boldsymbol{\theta}$, and its derivatives, and of the vector of generalized coordinates $\boldsymbol{q_0}$ and its derivatives; $\boldsymbol{\phi}$ is the vector of inertial parameters to estimate: $\boldsymbol{\phi} = \begin{bmatrix} \boldsymbol{\phi}_0^T & \boldsymbol{\phi}_1^T & \cdot\cdot & \boldsymbol{\phi}_N^T \end{bmatrix}^T$, $\boldsymbol{\phi}_j$ is the vector of standard parameters for each link B_j ($j = 0$ to N):

$$\boldsymbol{\phi}_j = [I_{xx,j} \quad I_{yy,j} \quad I_{zz,j} \quad I_{yz,j} \quad I_{zx,j} \quad I_{xy,j} \quad m_j \quad ms_{x,j} \quad ms_{y,j} \quad ms_{z,j}]^T \tag{3}$$

m_j is the mass, $I_{j,xx}$, $I_{j,yy}$, $I_{j,zz}$, $I_{j,yz}$, $I_{j,zx}$, $I_{j,xy}$ are the 6 components of the inertia matrix $\boldsymbol{I}_j$ at the joint center, $ms_{x,j}$, $ms_{y,j}$, $ms_{z,j}$ are the first moments components of $\boldsymbol{ms}_j$. Using only the upper part of (2) we obtain the reduced system given by (4) that is no longer function of the joint torque $\boldsymbol{\tau}$. (4) is as the classical identification problem. In the following, to simplify notations we will use $\boldsymbol{Y}_O$ to denote the contribution of the external forces on the base link.

$$\boldsymbol{W}_O\boldsymbol{\phi} = \sum_{k=1}^{N_c} \boldsymbol{K}_{kO}\boldsymbol{F}_k = \boldsymbol{Y}_O \tag{4}$$

For simplification, we write the general identification model as follows:

$$\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{X} + \boldsymbol{\rho} \tag{5}$$

3 Linear least squares identification technique (LS)

(5) after sampling can be written as follows, where $\boldsymbol{Y}_{ID}$ is the sampled vector of forces and torques $\boldsymbol{Y}$, and $\boldsymbol{W}_{ID}$ is the sampled regressor $\boldsymbol{W}$:

$$\boldsymbol{Y}_{ID} = \boldsymbol{W}_{ID}\boldsymbol{X} + \boldsymbol{\rho} \tag{6}$$

The identification problem consists in finding $\boldsymbol{X}$ that minimizes $\|\boldsymbol{\rho}\|$ or:

$$\| \boldsymbol{\rho} \|^2 = \| \boldsymbol{Y}_{ID} - \boldsymbol{W}_{ID}\boldsymbol{X} \|^2 \tag{7}$$

Base parameters and essential parameters

$\boldsymbol{X}$ is not identified directly when solving (6) as there is a rank deficiency of $\boldsymbol{W}_{ID}$ due to the kinematics structure of the system. The c columns of $\boldsymbol{W}$ are not independent: $rank(\boldsymbol{W}) = b$ such that $b \leq c$. Consequently, there exists an infinity of solutions for $\boldsymbol{\phi}$, only some of which are physically meaningful. Depending on the trajectory used for the identification, there may also be a rank deficiency due to the measurements, i.e. a lack of excitation. In that case, a numerical method to compute $\boldsymbol{\phi}_B$ is adequate Gautier (1991). It is based on the regressor matrix, it does not reflect the possibility that some parameters are almost 0, and have no contribution on the system; or that some parameters will be identified with a very low confidence. Removing these parameters from the model, thus considering them as no longer base parameters, allows to obtain a set of *essential parameters* sufficient to describe the system, taking into account measurement issues and model issues. The number e of *essential parameters* is such that $e \leq b < c$.

To compute the *essential parameters* we propose: (1) Compute $\boldsymbol{\phi}_B$, (2) Identify $\boldsymbol{\phi}_B$ using conventional method. Compute the standard deviation sx_i for each. sx_{max} is the highest standard deviation. (3) If the parameter which standard deviation is $sx_i = s_{max}$ if $s_{max} > \mu sx_{min}$. (4) Remove the column of $\boldsymbol{W}_B$ that correspond to the above selected parameter. Which means that it is no longer a base parameters. Reiterate the operation until $s_{max} \leq \mu sx_{min}$. Once the number of parameters to identify is defined, the identification problem can be solved. The solution is commonly found using a QR decomposition of the regressor matrix, the computation of the Moore Penrose pseudo inverse. Or using the SVD decomposition. In this paper we propose to calculate the LS solution of (6), in terms of the unique standard parameters that are the closest of a set of a-priori known values with the latter method, this ensures that the parameters are physically meaningful.

4 Inertial parameters identification with SVD factorization

SVD decomposition for identification

A solution of (6) can be obtained using a SVD decomposition. As we consider the standard identification model, from the c columns of $\boldsymbol{W}$ we make a distinction between the s ($s = b$ or $s = e$) columns that are used for identification and the $c - s$ columns that are used for calibration.

$$\boldsymbol{W}_{ID} \left[\boldsymbol{V}_1 \; \boldsymbol{V}_2\right] = \boldsymbol{U} \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{0}_{s,(c-s)} \\ \boldsymbol{0}_{(r-s),s} & \boldsymbol{0}_{(r-s),(c-s)} \end{bmatrix} \tag{8}$$

where $\boldsymbol{\Sigma} \in \boldsymbol{R}^{s \times b}$ is the diagonal matrix composed of the s singular values, $\boldsymbol{\Sigma} = diag(\sigma_i)$, with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_s \geq 0$, $\boldsymbol{V}_1 \in \boldsymbol{R}^{c \times s}$, $\boldsymbol{V}_2 \in \boldsymbol{R}^{c \times (c-s)}$, $\boldsymbol{U} \in \boldsymbol{R}^{r \times c}$. By multiplying respectively $\boldsymbol{Y}_{ID}$ and $\boldsymbol{W}_{ID}\boldsymbol{X}$ on the left by $\boldsymbol{U}^T$ we obtain (9) and (10).

$$\boldsymbol{U}^T \boldsymbol{Y}_{ID} = \boldsymbol{G} = \begin{bmatrix} \boldsymbol{U}_1^T \boldsymbol{Y}_{ID} \\ \boldsymbol{U}_2^T \boldsymbol{Y}_{ID} \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_1 \\ \boldsymbol{G}_2 \end{bmatrix} \tag{9}$$

$$\boldsymbol{U}^T \boldsymbol{W}_{ID} \boldsymbol{X} = \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{0}_{s,(c-s)} \\ \boldsymbol{0}_{(r-s),s} & \boldsymbol{0}_{(r-s),(c-s)} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_1^T \boldsymbol{X} \\ \boldsymbol{V}_2^T \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma} \boldsymbol{V}_1^T \boldsymbol{X} \\ \boldsymbol{0}_{(c-s),1} \end{bmatrix} \tag{10}$$

And by defining $\boldsymbol{Z}$ such that $\boldsymbol{Z} = \boldsymbol{V}^T \boldsymbol{X} = \begin{bmatrix} \boldsymbol{V}_1^T \boldsymbol{X} \\ \boldsymbol{V}_2^T \boldsymbol{X} \end{bmatrix} = \begin{bmatrix} \boldsymbol{Z}_1 \\ \boldsymbol{Z}_2 \end{bmatrix}$. We also have: $\boldsymbol{X} = \boldsymbol{V}\boldsymbol{Z}$. Finally, for all $\boldsymbol{Z}_2$ we obtain:

$$\boldsymbol{U}^T \boldsymbol{W}_{ID} \boldsymbol{X} \;\; = \;\; \begin{bmatrix} \boldsymbol{\Sigma} \boldsymbol{Z}_1 \\ \boldsymbol{0}_{(c-s),1} \end{bmatrix} \tag{11}$$

$\boldsymbol{U}$ keeps the norm unchanged, the identification problem (7) is then similar to the following equation:

$$\begin{aligned} \| \boldsymbol{\rho} \|^2 &= \| \boldsymbol{Y} - \boldsymbol{W}_{ID}\boldsymbol{X} \|^2 \\ &= \| \boldsymbol{U}^T\boldsymbol{Y}_{ID} - \boldsymbol{U}^T\boldsymbol{W}_{ID}\boldsymbol{X} \|^2 \\ &= \| \boldsymbol{G}_1 - \boldsymbol{\Sigma}\boldsymbol{Z}_1 \|^2 + \| \boldsymbol{G}_2 \|^2 \end{aligned} \tag{12}$$

$\hat{\boldsymbol{Z}}_1$ is the unique solution of $\boldsymbol{\Sigma}\hat{\boldsymbol{Z}}_1 = \boldsymbol{G}_1$. $\boldsymbol{\Sigma}$ is diagonal, consequently $\hat{\boldsymbol{Z}}_1$ is given by $\hat{\boldsymbol{Z}}_1 = \boldsymbol{\Sigma}^{-1}\boldsymbol{G}_1 = \boldsymbol{\Sigma}^{-1}\boldsymbol{U}_1^T\boldsymbol{Y}_{ID}$. We deduce the family of optimal solutions, for any $\boldsymbol{Z}_2$:

$$\hat{\boldsymbol{Z}} = \begin{bmatrix} \hat{\boldsymbol{Z}}_1 \\ \boldsymbol{Z}_2 \end{bmatrix} \tag{13}$$

Finally the optimal solution $\hat{\boldsymbol{X}}$ to (7) is given by: $\hat{\boldsymbol{X}} = \boldsymbol{V}\hat{\boldsymbol{Z}} = \boldsymbol{V}_1\boldsymbol{\Sigma}^{-1}\boldsymbol{U}_1^T\boldsymbol{Y}_{ID} + \boldsymbol{V}_2\boldsymbol{Z}_2$. The norm of the error, for any optimal solution is given by $\| \boldsymbol{\rho} \|_{min} = \| \boldsymbol{G}_2 \| = \| \boldsymbol{U}_2^T\boldsymbol{Y} \|$. We obtain the unique solution $\boldsymbol{X}_{opt}$ that minimizes the error: $\| \hat{\boldsymbol{X}} \| = \| \hat{\boldsymbol{Z}} \|$ for $\boldsymbol{Z}_2 = \boldsymbol{0}_{(c-b),1}$, and:

$$\boldsymbol{X}_{opt} = \boldsymbol{V}_1\boldsymbol{\Sigma}^{-1}\boldsymbol{U}_1^T\boldsymbol{Y}_{ID} \tag{14}$$

Among the optimal solutions $\hat{\boldsymbol{X}}$ that minimizes $\|\boldsymbol{\rho}\|$, the Moore Penrose pseudo inverse $\hat{\boldsymbol{X}}_{mp}$ minimizes $\|\hat{\boldsymbol{X}}\|$, with $\boldsymbol{Z}_2 = \boldsymbol{0}_{(c-b),1}$.

Calibration using SVD decomposition

The solution obtained by (14) is optimal in term of $\|\boldsymbol{\rho}\|$; however the consistency of the parameters is not guaranteed. The problem of the physical consistency has been addressed recently in Yoshida and Khalil (2000), Nakanishi and et al. (2008), Diaz-Rodriguez and et al. (2010) and Ayusawa and Nakamura (2010), by adding constraints to the system so that the inertia matrices are definite positive, the masses are positive and eventually that the center of mass is located into a convex hull. These results are not optimal and $\hat{\boldsymbol{X}}$ does not minimize $\|\boldsymbol{\rho}\|$. Moreover, the identification requires a lot of recorded experimental data (40 minutes) Nakanishi and et al. (2008). We propose another approach. Using physically consistent a-priori values for $\boldsymbol{X}$, denoted $\boldsymbol{X}^{ref}$, such that:

$$\boldsymbol{Y}_{ID}^{ref} = \boldsymbol{W}_{ID}\boldsymbol{X}^{ref} \tag{15}$$

We propose to find the optimal solution that calibrates the system (16) obtained by the substraction of (5) and (15), using the SVD decomposition.

$$\boldsymbol{Y}_{ID} - \boldsymbol{Y}_{ID}^{ref} = \boldsymbol{W}_{ID}(\boldsymbol{X} - \boldsymbol{X}^{ref}) + \boldsymbol{\rho} \tag{16}$$

By substitution in (14) we obtain:

$$\boldsymbol{X}_{opt} = \boldsymbol{V}_1 \boldsymbol{\Sigma}^{-1} \boldsymbol{U}_1^T (\boldsymbol{Y}_{ID} - \boldsymbol{Y}^{ref}) + \boldsymbol{X}^{ref} = \boldsymbol{X}_{mp} + \boldsymbol{V}_2 * \boldsymbol{V}_2^T \boldsymbol{X}^{ref} \quad (17)$$

Finally, the solution of (17) is the least square solution using the Moore Penrose pseudo inverse $\boldsymbol{X}_{mp}$ and the projection in the null space of the regressor of the a priori parameters $\boldsymbol{X}^{ref}$. It minimizes the same residual norm $\| \boldsymbol{\rho} \|$ given by (7), of the residual $\boldsymbol{\rho}$ in (6). It minimizes at the same time $\|\boldsymbol{X} - \boldsymbol{X}^{ref}\|$ in the sense of the Euclidian norm. $\boldsymbol{X}$ preserves the properties of $\boldsymbol{X}^{ref}$, in particular the physical consistency.

5 Application to the human body

Measurement and modeling

We identify the SP of the human body. Measuring the SP in-vivo is complicated and often inaccurate. Yet, a precise knowledge of these information is crucial for motion studies. The human body is modelled using a kinematic chain made of 15 rigid links and 40 DOF. We use the root-link equation to perform the identification, the joint torques are not needed. Thus (17) is now obtained by the SVD decomposition of $\boldsymbol{W}_O$ instead of $\boldsymbol{W}$ (respectively $\boldsymbol{W}_{OID}$ instead of $\boldsymbol{W}_{ID}$), and $\boldsymbol{Y}$ is replaced by $\boldsymbol{Y}_O$ (respectively $\boldsymbol{Y}_{OID}$ instead of $\boldsymbol{Y}_{ID}$), as shown in (18).

$$\boldsymbol{X}_{opt} = \boldsymbol{V}_1 \boldsymbol{\Sigma}^{-1} \boldsymbol{U}_1^T (\boldsymbol{Y}_{OID} - \boldsymbol{Y}_{OID}^{ref}) + \boldsymbol{X}^{ref} \quad (18)$$

The motion and force data is captured by a motion captured system. Inverse kinematics gives the joint angles, velocities and accelerations, from which $\boldsymbol{W}_{OID}$ and $\boldsymbol{Y}_{\boldsymbol{OID}}$ are computed. *persistent exciting trajectories* by Gautier and Khalil (1992)are obtained using our previous works Venture et al. (2009b), Venture et al. (2009a).

Experimental results

Because it is a calibration procedure, the choice of the reference $\boldsymbol{X}^{ref0}$ is crucial. We use the result obtained from Ayusawa and Nakamura (2010) and noted $\boldsymbol{X}^{ref0}$. The calibrated parameters using this reference are $\boldsymbol{X}^{0b}$ when using the base parameters; and $\boldsymbol{X}^{0e}$ when using the essential parameters. We verify the physical consistency, i.e. the mass is positive and the inertia matrix at the center of mass (CoM) $\boldsymbol{J}_j$ is positive definite. The latter can be tested by eigenvalue decomposition, the Sylvester theorem, or Cholesky decomposition. They are equivalent, however, the Sylvester theorem allows

Table 1. Identified standard parameters

Lower torso	m	ms_x	ms_y	ms_z	J_{xx}	J_{yy}	J_{zz}	J_{xy}	J_{xz}	J_{yz}
$\boldsymbol{X}^{ref0}$	21.44	-0.63	0.04	4.04	0.10	0.04	0.12	0.000	0.004	-0.000
$\boldsymbol{X}^{0b}$	21.81	-0.98	0.02	3.05	-0.09	0.25	0.11	-0.05	0.11	-0.01
$\boldsymbol{X}^{0e}$	21.44	-0.63	0.03	4.04	0.10	0.04	0.12	0.000	0.007	-0.001
Right forearm	m	ms_x	ms_y	ms_z	J_{xx}	J_{yy}	J_{zz}	J_{xy}	J_{xz}	J_{yz}
$\boldsymbol{X}^{ref0}$	1.51	0.14	0.043	-0.025	0.001	0.002	0.004	0.000	0.000	-0.000
$\boldsymbol{X}^{0b}$	1.36	0.15	0.029	-0.004	-0.01	0.005	0.03	0.004	-0.05	-0.02
$\boldsymbol{X}^{0e}$	1.51	0.13	0.038	-0.029	0.000	0.007	0.01	0.003	0.002	-0.001

finding conditions that the parameters must verify to obtain the positive definitiveness Yoshida and Khalil (2000). These conditions help adjusting the parameters to obtain a positive definite matrix by modifying the SP from the null space of the regressor, i.e. the non base parameters. The parameters are not independent, thus modifying one parameter results in modifying all the non base parameters. Manipulations need precautions. The Cholesky decomposition presents the advantage that a tolerance $\varepsilon \leq 0$ can be set allowing for taking into account noise and measurement error. It is similar to setting a numerical zero in the SVD or QR decomposition. ε is chosen according to the error and the level of noise in the data. $\boldsymbol{X}^{ref0}$ is to chosen to be physically consistent strictly ($\varepsilon = 0$). The identified parameters and the reference for the lower torso and the right forearm are given as examples in Table 1. The parameters obtained with the base parameters: $\boldsymbol{X}^{0b}$ leads to a on-definite positive inertia matrix of the lower torso and the right forearm (with a tolerance $\varepsilon = -0.05$). The essential parameters leads to definitive positive matrices for all the links with the same tolerance.

6 Conclusion

Dynamics identification based on robotics formalism, motion information and contact force information, provides a simple method to estimate the SP accurately. We presented a solution to calibrate the SP. The obtained solution preserves the minimal residual norm; and with an appropriate choice for the initial value, we also guaranty the physical consistency. The method is based on the singular value decomposition (SVD) of the regressor matrix. We applied our algorithm to experimental data recorded using motion capture data and force plate information to identify the human whole body mass parameters: 150 SP. We proved that the proposed method gives an optimal solution for the SP. With small measurement error it is physically meaningful and it minimizes the norm of the residual. These results are obtained thanks to the choice of an appropriate reference vector of SP.

Bibliography

K. Ayusawa and Y. Nakamura. Identification of standard inertial parameters for large-dof robots considering physical consistency. *Proc. Int. Conf. on Intelligent Robots and Systems*, pages 6194–6201, 2010.

C.K. Cheng. Segment inertial properties of chinese adults determined from magnetic resonance imaging. *Clinical biomechanics*, 15:559–566, 2000.

M. Diaz-Rodriguez and et al. A methodology for dynamic parameters identification of 3-dof parallel robots in terms of relevant parameters. *Mechanism and Machine Theory*, pages 1337–1356, 2010.

R. Dumas and et al. Adjustments to mcconville et al. and young et al. body segment inertial parameters. *J. Biomechanics*, 40:543–553, 2007.

Y. Fujimoto and et al. Robust biped walking with active interaction control between foot and ground. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, page 20302035, 1998.

M. Gautier. Numerical calculation of the base inertial parameters. *J. of Robotic Systems*, 8(4):485–506, 1991.

M. Gautier and W. Khalil. Exciting trajectories for the identification of base inertial parameters of robots. *Int. J. of Robotic Research*, 11(4): 363–375, 1992.

W. Khalil and E. Dombre. *Modeling, identification and control of robots*. Hermès Penton, London-U.K, 2002.

O. Khatib and et al. Whole-body dynamic behavior and control of humanoid-like robots. *Int. J. of Humanoid Robotics*, 1(1):29–43, 2003.

M.J. Mataric. Learning social behavior. *Robotics and Autonomous Systems*, (20):191–204, 1997.

J. Nakanishi and et al. Operational space control: A theoretical and empirical comparison. *Int. J. of Robotics Research*, (27):737–757, 2008.

G. Venture, K. Ayusawa, and Y. Nakamura. Real-time identification and visualization of human segment parameters. In *Proc. of the IEEE Int. Conf. on Engineering in Medicine and Biology*, pages 3983–3986, 2009a.

G. Venture, K. Ayusawa, and Y. Nakamura. A numerical method for choosing motions with optimal excitation properties for identification of biped dynamics - an application to human. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 1226–1231, 2009b.

K. Yoshida and W. Khalil. Verification of the positive definiteness of the inertial matrix of manipulators using base inertial parameters. *Int. J. of Robotics Research*, pages 498–510, 2000.

Assessment of Physical Exposure to Musculoskeletal Risks in Collaborative Robotics Using Dynamic Simulation

Pauline Maurice * and Yvan Measson * and Vincent Padois † and Philippe Bidaud †

* CEA LIST, Fontenay-aux-Roses, France
† ISIR, University Pierre and Marie Curie, Paris, France

Abstract Many industrial tasks cannot be executed by a robot alone. A way to help workers in order to decrease the risk of musculoskeletal disorders is to assist them with a collaborative robot. Yet assessing its usefulness to the worker remains costly because it usually requires a prototype. We propose a dynamic simulation framework to model the performing of a task jointly by a virtual manikin and a robot. It allows to measure physical quantities in order to perform an ergonomic assessment of the robot. Experiments are carried out on two different robots. The results show that the proposed simulation framework is helpful for designing collaborative robots. Further work includes enhancing the simulation realism and validation on a real robot.

1 Introduction

Though working conditions have improved in developed countries, work-related musculoskeletal disorders (MSD) remain a major health problem. These disorders affect the body's muscles, joints, tendons, ligaments and nerves. They result from strenuous biomechanical solicitations caused by physical work. According to several studies (Silverstein and Adams, 2005, and Jones et al., 2005), over 50% of workers in industry suffer from MSD. In France it represents about 80% of the occupational diseases in 2009 (Ha and Roquelaure, 2010). Indeed, despite the growing robotization in industry, many hard tasks cannot be fully automatized, because of their unpredictability or their technicality. A solution is to assist the worker with a collaborative robot (or intelligent assist device), rather than replacing him. A collaborative robot enables the joint manipulation of objects with the worker and thereby provides a variety of benefits such as strength amplification, inertia masking and guidance via virtual surfaces and paths (Colgate et al., 2003).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_40,

The purpose of these collaborative robots is to decrease the risk of MSD by alleviating the worker's physical load and improving his posture. One of the main issues in the design process of a collaborative robot is to take into account the human presence and capabilities. Yet performing an ergonomic assessment of such a robot is essential to check its usefulness to the worker. Many methods exist (Guangyan and Buckle, 1999), based on the observation of an actual worker, but they need a prototype of the robot. It is a significant limitation in terms of cost and time. An alternative is to perform the assessment within a digital world, using a virtual manikin to simulate the worker. Digital human models are already available to evaluate the design of workstations, such as JACK, RAMSIS, SAFEWORK or SAMMIE (Blanchonette, 2010, and Porter et al., 2004). But they do not allow a fully automatic and dynamic simulation of realistic movements. Moreover, the commercial software frameworks in which these manikins are integrated were not designed with collaborative robotics in mind.

We propose a method based on a dynamic simulation framework including a digital manikin, to estimate *a priori* the biomechanical benefit of a collaborative robot. This framework has already been used for a work task ergonomic assessment in De Magistris et al. (2011). Our contribution regards the addition of a collaborative robot interacting with the manikin, so that they perform the task jointly. Section 2 describes the simulation framework and the test case. Section 3 details the results and discusses their relevance towards the MSD risk assessment. Section 4 presents the conclusions and perspectives of this study.

2 Materials and Methods

The purpose of the simulation is to have a virtual manikin automatically perform a given work task, with and without the help of a collaborative robot, in order to measure biomechanical quantities (see Figure 1 for the description of the simulation framework). The manikin is described in Section 2.1, the robot and the task in Section 2.2 and Section 2.3.

2.1 Virtual Manikin

The human body is modelled kinematically as an arborescence of rigid bodies linked by hinge joints. The actuation model is articular (there is no muscle model) and the root is not controlled.

By modifying the worker's efforts needed to perform the task, the use of a collaborative robot affects the worker's posture. So data from real workers cannot be used to control the manikin. The manikin posture is

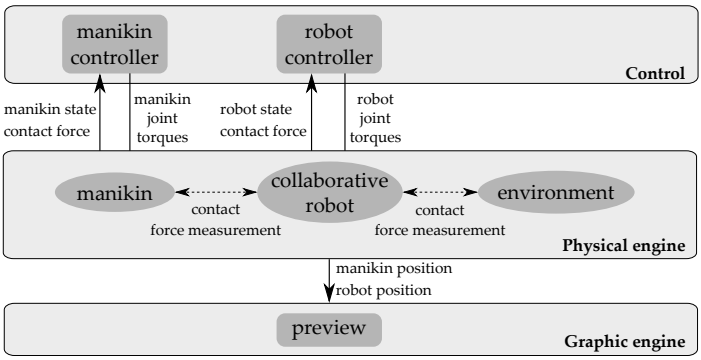


Figure 1. Description of the simulation framework.

generated at each time step thanks to a multi-objective optimization with a force control approach based on a quadratic programming problem and a Jacobian transpose control method proposed by Liu et al. (2011). The different objectives are :

- Center of mass: the horizontal position of the CoM is controlled to maintain the manikin balance. The CoM vertical position is controlled to keep the manikin upright.
- Contact forces: to produce stable motion with smaller contact forces, one of the objectives is to minimize the contact forces between the feet and the ground.
- Manipulation task: the position and orientation of the right hand are controlled to follow a trajectory depending on the task. The contact forces on the hand can also be controlled to match a reference effort.
- Visibility: the orientation of the head is controlled to direct the face of the worker towards the work zone.
- Posture: a secondary articular objective is added because of the redundancy of the human body. The manikin posture should be as close as possible to the one of a man resting upright.

2.2 Collaborative Robot

The robot was designed by the CEA and the RB3D company to help workers in manual tasks which require the application of significant efforts via a portable tool. It provides strength amplification (Kazerooni, 1990). The robot A is a serial chain with 6 hinge joints (see Figure 2 left). The tool is attached to the tip of the end effector and the robot is manipulated by the worker via a handle fixed on the end effector. Another version (robot B)

of this robot was also tested as a comparison. Two joint axes are inverted, and some segments lengths are modified (see Figure 2 right).

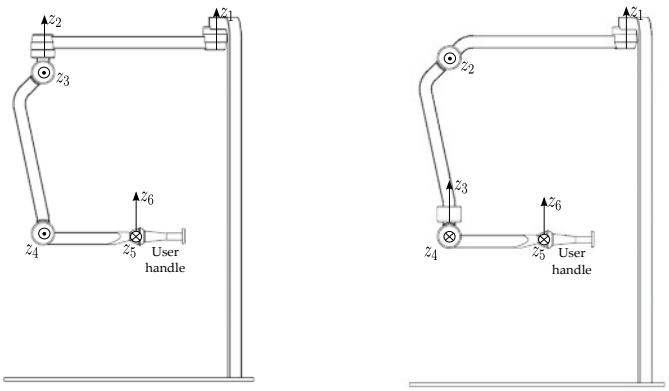


Figure 2. Kinematic architecture of collaborative robots A (left) and B (right).

The control of the robot consists in compensating for gravity and dry and viscous friction, and amplifying the efforts of the worker. The force applied on the tool is equal to the one applied by the worker multiplied by a coefficient α. A force sensor on the user handle allows the force measurement. The real control framework is simplified since there is no actuator model (one controls directly the joint torques) and the dry friction is not yet implemented in the physical engine of the simulation (see Figure 3).

2.3 Task

The purpose of these first tests is more to validate the assessment method than to evaluate a collaborative robot. So a simple task was chosen. The manikin moves the tool back and forth between point A and point B, and stays 4s on each point. Point A is located on the surface of a fixed rigid body. There the normal contact force must be kept to a stable value ($80N \pm 5\%$). Point B is 10cm backwards.

3 Results and Discussion

The main biomechanical risk factors for MSD are extreme postures, considerable efforts, high frequency of the gestures and static work. Therefore we are particular interested in the joints angles and torques. The shoulder

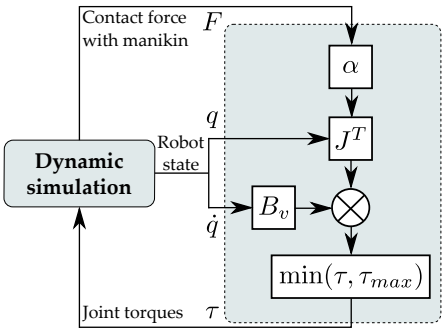


Figure 3. Block diagram of the robot control framework.

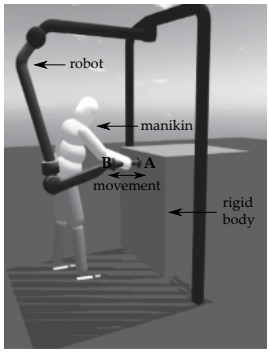


Figure 4. Screenshot of the simulation.

joint for arm flexion and extension is the most solicited one in the test task, and the wrist joint has very limited effort capacities, so only their results are presented.

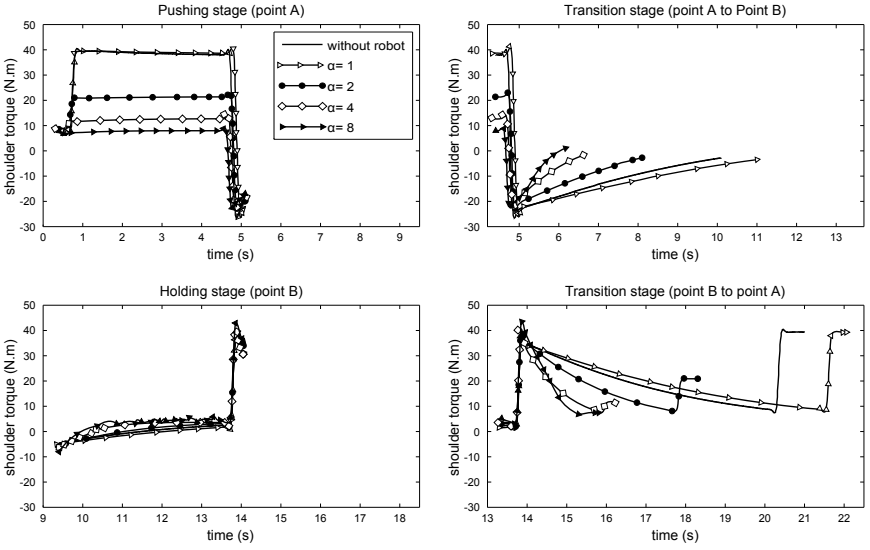


Figure 5. Shoulder torque for robot A. The duration of the transition stages varies with α, so the graphs are divided into 4 parts and synchronized to be more understandable.

3.1 Collaborative Robot A

We simulated the task with and without robot A, and for various values of the amplification coefficient α. First we notice that the transition stages (point A to point B and *vice versa*) are much quicker when α increases (see Figure 5 bottom left and bottom right). The time needed to perform the task is reduced, without needing extra strength from the manikin.

During the pushing stage (point A) the expected decrease in the internal efforts with the amplification is observed (see Figure 5 top left). Though it is hard to directly link the torque values to an objective risk of MSD, they could be a first hint on how to estimate the benefit of a collaborative robot.

During the holding stage (point B) oscillations occur when $\alpha = 8$ (see Figure 5 top right). It means that the amplification is too important to ensure the robot stability.

However the results without assistance and with a non amplified robot ($\alpha = 1$) are too similar to be realistic. The transparency of the robot is practically perfect. This is partly due to the lack of dry friction in the simulation, which cannot be fully compensated on a real robot.

We simulated the same task (with $\alpha = 2$) with faster movements and we varied the robot mass. The peak torque needed to accelerate and slow down the robot decreases with the mass (see Table 1): the inertial effects cannot be fully compensated by the robot control. So the choice of the robot mass is critical.

mass	accelerate	slow down
$0.1 * m_0$	57.5	-30.9
$1 * m_0$	67.8	-35.0
$2 * m_0$	72.1	-37.6
$4 * m_0$	85.3	-44.8

Table 1. Peak shoulder torque (N.m) during the transition stages, for different masses.

3.2 Collaborative Robot B

A second experiment was carried out, with robot B. Unlike robot A, during the pushing and holding stages the shoulder angle and torque are much higher with robot B than without it, even with strength amplification (see Table 2 and 3). According to the EU norm AFNOR (2008), even with no external load, the bigger the angle, the higher the risk of MSD.

During the transition stages, the wrist is much more solicited with robot B than with robot A (see Table 4). Because of its kinematic, more efforts

are needed to prevent the end effector of robot B from moving right or left while going forward and backward.

Although these two collaborative robots are quite alike, robot A improves the worker situation, whereas robot B is not suitable for this task.

	push	hold
without robot	30.4	17.0
robot A $\alpha = 1$	31.1	17.7
robot B $\alpha = 1$	35.9	30.1
robot A $\alpha = 4$	29.4	14.8
robot B $\alpha = 4$	38.9	30.2

Table 2. Mean shoulder angle (deg) during the pushing and holding stages, for robot A and B.

	push	hold
without robot	38.5	1.0
robot A $\alpha = 1$	39.1	0.1
robot B $\alpha = 1$	52.6	10.2
robot A $\alpha = 4$	12.4	3.3
robot B $\alpha = 4$	23.6	12.3

Table 3. Mean shoulder torque (N.m) during the pushing and holding stages, for robot A and B.

	difference (%)
without robot	0.0
robot A $\alpha = 1$	+1.2
robot B $\alpha = 1$	+18.4
robot A $\alpha = 4$	-13.0
robot B $\alpha = 4$	+18.5

Table 4. Comparison of peak wrist torque with and without robot, during the acceleration stage, for robot A and B.

4 Conclusion and Future Work

We proposed a method to carry out an ergonomic assessment of a collaborative robot, based on a dynamic simulation framework and a digital manikin. We simulated the performing of a simple task by the manikin, alone and assisted by two collaborative robots with various force amplification. One of the robots induces a significant decrease in the manikin efforts. The other one leads to a situation which is worse than without assistance. This proves the usefulness of this framework, which avoids building useless prototypes and helps to tune some parameters of the robot.

The first robot is currently being built, so the results will now have to be validated on the real robot. Further work is directed towards the enhancement of the simulation realism. The modelling of dry friction has

to be implemented, because this phenomenon significantly affects the efforts needed to move the robot. The control framework of the manikin will be improved, so that its behaviour is more human-like. In particular, the posture adaptation depending on the external load is critical, since the posture is one of the major risk factors for MSD.

Bibliography

AFNOR. NF EN 1005-4+A1, Safety of machinery - Human physical performance. French standard institute (AFNOR) - European committee for standardization, november 2008.

P. Blanchonette. Jack human modelling tool: A review. Technical report, DTIC Document, 2010.

J.E. Colgate, M. Peshkin, and S.H. Klostermeyer. Intelligent assist devices in industrial applications: a review. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2516–2521. IEEE, 2003.

G. De Magistris, A. Micaelli, C. Andriot, J. Savin, and J. Marsot. Dynamic virtual manikin control design for the assessment of the workstation ergonomy, 2011.

LI Guangyan and P. Buckle. Current techniques for assessing physical exposure to work-related musculoskeletal risks, with emphasis on posture-based methods. *Ergonomics*, 42(5):674–695, 1999.

C. Ha and Y. Roquelaure. Troubles musculo-squelettiques d'origine professionnelle en France. Où en est-on aujourd'hui ? *Bulletin épidémiologique hebdomadaire*, (5-6):35–36, fvrier 2010.

J.R. Jones, C.S. Huxtable, and J.T. Hodgson. Self-reported work-related illness in 2004/05. Technical report, Health, Safety and Executive, 2005.

H. Kazerooni. Human-robot interaction via the transfer of power and information signals. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):450–463, 1990.

M. Liu, A. Micaelli, P. Evrard, A. Escande, and C. Andriot. Interactive dynamics and balance of a virtual character during manipulation tasks. In *Robotics and Automation, 2011. Proceedings. ICRA'11 IEEE International Conference on*. IEEE, 2011.

J.M. Porter, K. Case, R. Marshall, D. Gyi, and R. Sims Neé Oliver. Beyond Jack and Jill: designing for individuals using HADRIAN. *International Journal of Industrial Ergonomics*, 33(3):249–264, 2004.

B. Silverstein and D. Adams. Workrelated musculoskeletal disorders of the neck, back, and upper extremity in Washington State, 1995-2003. Technical report, Safety and Health Assessment and Research for Prevention (SHARP), Washington State Department of Labor and Industries, 2005.

Muscle Activity Estimation Based on Inverse Dynamics and Muscle Stress Analysis by Finite Element Method

Kensho Hirasawa† *, Ko Ayusawa† , and Yoshihiko Nakamura†

† Department of Mechano-Informatics, The University of Tokyo, Tokyo, Japan

Abstract This paper proposes a method to estimate muscle activity with taking account the volumetric effects of muscles. We analyze muscles as elastic bodies by using finite element method combined with computation technique in robotics. We describe the way to estimate muscle force and deformation as the problem to find muscle activity which produce joint torque calculated by inverse dynamics subject to equilibrium equation of elastic bodies. This problem is solved as sequential quadratic programming with finite element analysis in parallel for each muscle.

1 Introduction

Many researchers have proposed mechanical models of human body based on physiological and anatomical knowledge to clarify the mechanism of human motion. Some researches focus on a specific part of human body. Though their models are a set of few bones and muscles, each component is often analyzed precisely with finite element method (FEM). For example, Bachtar et al. (2006) estimated stress distribution on hip joint by using contact analysis based on FEM. Another example is the simulation of muscle contraction considering effect of change of muscle shape (T. Johansson, P. Meier and R. Blickhan (2000)). FEM enables modeling of arbitrary geometry and detailed analysis. However, finite element analysis (FEA) of many components generally requires high computational effort.

On the other hand, computation technique developed in robotics is applied for whole-body analysis of human. Y. Nakamura, K. Yamane and I. Suzuki (2005) and K. Yamane, Y. Fujita and Y. Nakamura (2005) modeled human body as rigid bodies and wires and estimated muscle activity during whole-body motion with robotics computation. S. L. Delp, F. C. Anderson,

* This research is supported by HPCI STRATEGIC PROGRAM Computational Life Science and Application in Drug Discovery and Medical Development by MEXT.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_41,

A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman and D. G. Thelen (2007) also developed a simulator for human motion analysis with a wire-driven multi-body model of whole body. By modeling muscles as wires, these researches analyzed whole body which consists of hundreds of components. However, these models are difficult to be taken account of effects caused by change of three dimensional deformation of muscles. Consideration of contact force between components is also another problem in the models.

Our research purpose is to analyze human motion in detail considering change of muscle force caused by the deformation of muscles. To consider the effect by deformation, we model muscles as three dimensional elastic bodies and utilize FEM. For the reduction of computation complexity, bones are modeled as rigid bodies and computation technique in robotics is applied. Since bones are assumed to be not deformed, FEA of a muscle becomes independent of the other muscles. This allows us to reduce computation complexity by parallel computing.

2 Muscle activity estimation using finite element analysis and inverse dynamics of multi-body system

2.1 Muscular skeletal model

We model body segments as rigid bodies and muscles as three dimentional elastic bodies. Elastic bodies are represented as finite element mesh and analyzed by FEM. The basic concept of this muscular skeletal model is presented in Fig.1. We assume Saint Venant-Kirchhoff material as the material model of elastic bodies. Saint Venant-Kirchhoff material has the linear relationship between strain and stress while the relationship between displacement and strain is nonlinear.

We separate elastic bodies into the regions which generate muscle contraction force and the regions corresponding to tendons and ligaments which only generate passive stress. The contraction force caused by muscle activation is modeled as continuous contraction stress generated by uniform muscle activity. The initial direction of muscle fiber is defined at nodal points and interpolated at other points. If muscles are deformed, the direction is recomputed based on the deformation.

We use muscle force model proposed by T. Johansson, P. Meier and R. Blickhan (2000). In this model, the absolute value p of the component in muscle fiber direction of Cauchy stress generated by muscle activation is given by the following equations.

$$p = a f_l f_v p_{iso} \beta \tag{1}$$

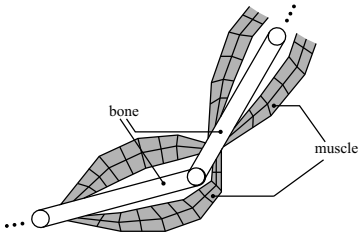


Figure 1. Muscular skeletal model which consists of elastic bodies as muscles and rigid bodies as bones.

where, a is muscle activity, f_l is coefficient relevant to force-length relationship of muscle, f_v is coefficient relevant to force-velocity relationship of muscle and p_{iso} is the maximum muscle stress in isometric contraction. β is the ratio of cross sectional area in muscle fiber direction in undeformed state to that in deformed state. The term β makes the maximum muscle contraction force in infinitesimal region constant even during deformation. This reflects that the number of muscle fiber in a certain region is constant during deformation. We use force-length relationship f_l and force-velocity relationship f_v proposed by A. V. Hill (1938) and J. L. van Leeuwen (1991). the coefficients of these relationship are derived from muscle deformation.

Edges of muscles connect to bones. Joint angle on multi-body system of bones gives the displacement boundary condition on the edges of muscles. In order to calculate force or joint torque, we have to calculate deformation of elastic bodies which satisfy the boundary condition. We assume that the change of joint angle and muscle force is small enough when duration is small, and compute deformation quasistatically. The deformation of a elastic body is calculated by FEM. In this paper, we compute the deformation by finding the deformation which minimize total potential energy of the elastic body (O. C. Zienkiewicz, R. L. Taylor and J. Z. Zhu (2005)).

2.2 Method of muscle activity estimation

We mention the method to estimate muscle activity from time-series data of joint angle. In this paper, we describe muscle activity estimation as the problem to minimize the difference between joint torque calculated by analysis of elastic bodies and that by inverse dynamics of multi-body

system. Let this problem be

$$\min_{\boldsymbol{a}} \Big(\frac{1}{2} \big\| \boldsymbol{\tau}(\boldsymbol{d}) - \boldsymbol{\tau}^* \big\|^2 + \frac{1}{2} w \|\boldsymbol{a}\|^2 \Big) \tag{2}$$

$$\text{s.t.} \quad \frac{\partial Z_i}{\partial \boldsymbol{d}_i}(a_i, \boldsymbol{d}_i) = \boldsymbol{0} \ , \tag{3}$$

$$0 \leq a_i \leq 1 \ (i = 1,\ 2, \cdots, N_m) \tag{4}$$

where, $\boldsymbol{a}$ is a vector of muscle activities and $\boldsymbol{d}$ is a vector of nodal displacement of finite element meshes. $\boldsymbol{\tau}$ stands for the joint torque evaluated by FEA from the displacement $\boldsymbol{d}$, and $\boldsymbol{\tau}^*$ stands for the reference of joint torque calculated by inverse dynamics from joint angle data. The second term of Eq.(2) has the purpose to find as small muscle activation which generate required joint torque as possible. w is weight for this term. The two constraints are applied for each muscle and N_m means the number of all muscles. Z_i means the total potential energy of a elastic body corresponding to i th muscle and Eq.(3) is a equilibrium equation of elastic bodies.

According to Eq.(3), we can obtain displacement $\boldsymbol{d}$ from muscle activity $\boldsymbol{a}$. Therefore, the objective function can be treated as a function of muscle activity $\boldsymbol{a}$. Furthermore, let this optimization problem be expressed as the following sequential quadratic problem.

$$\min_{\Delta \boldsymbol{a}} \Big(\frac{1}{2} \big\| \boldsymbol{\tau}^{(k)} + \frac{\partial \boldsymbol{\tau}^{(k)}}{\partial \boldsymbol{a}} \Delta \boldsymbol{a} - \boldsymbol{\tau}^* \big\|^2 + \frac{1}{2} w \|\boldsymbol{a}^{(k)} + \Delta \boldsymbol{a}\|^2 \Big) \tag{5}$$

$$\text{s.t.} \quad -a_i^{(k)} \leq \Delta a_i \leq 1 - a_i^{(k)} \ (i = 1,\ 2, \cdots, N_m) \tag{6}$$

where, k represents the number of iterations. $\boldsymbol{\tau}^{(k)}$ and $\partial \boldsymbol{\tau}^{(k)} / \partial \boldsymbol{a}$ stand for $\boldsymbol{\tau}$ and $\partial \boldsymbol{\tau} / \partial \boldsymbol{a}$ when the muscle activity is $\boldsymbol{a}^{(k)}$. To find muscle activity which minimize Eq.(2), we update muscle activity $\boldsymbol{a}$ using the obtained $\Delta \boldsymbol{a}$ according to the quadratic programming and a proper step size. Fig.2 shows the outline of the muscle activity estimation. We need to calculate joint torque $\boldsymbol{\tau}$ and the gradient of it to muscle activity $\partial \boldsymbol{\tau} / \partial \boldsymbol{a}$ for the quadratic programming as presented in Eq.(5) and Eq.(6). Each muscle can be analyzed in parallel and then we compute τ and $\partial \boldsymbol{\tau} / \partial \boldsymbol{a}$. For getting $\boldsymbol{\tau}$, we first calculate deformation $\boldsymbol{d}_i$ and derive force $\boldsymbol{f}_i$ and moment $\boldsymbol{m}_i$ of each muscle from $\boldsymbol{d}_i$. We then compute $\boldsymbol{\tau}$ by multiplying basic jacobian matrix by $\boldsymbol{f}_i$ and $\boldsymbol{m}_i$. The calculation of $\boldsymbol{d}_i$, $\boldsymbol{f}_i$ and $\boldsymbol{m}_i$ can be calculated in parallel for each muscle. $\partial \boldsymbol{\tau} / \partial \boldsymbol{a}$ is derived in the similar way to the derivation of $\boldsymbol{\tau}$. We first compute $\partial \boldsymbol{d}_i / \partial \boldsymbol{a}$, then compute $\partial \boldsymbol{f}_i / \partial \boldsymbol{a}$ and $\partial \boldsymbol{m}_i / \partial \boldsymbol{a}$ and derive $\partial \boldsymbol{\tau} / \partial \boldsymbol{a}$. $\partial \boldsymbol{d}_i / \partial \boldsymbol{a}$ can be computed numerically by solving the derivative equation of Eq.(3) with respect to muscle activity.

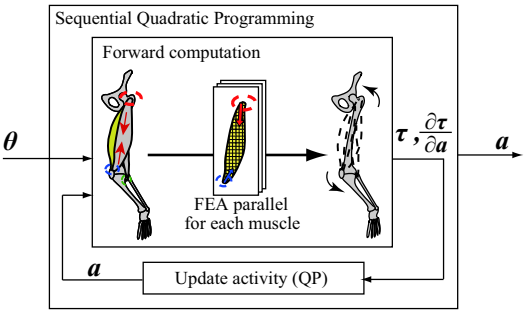


Figure 2. Outline of muscle activity estimation.

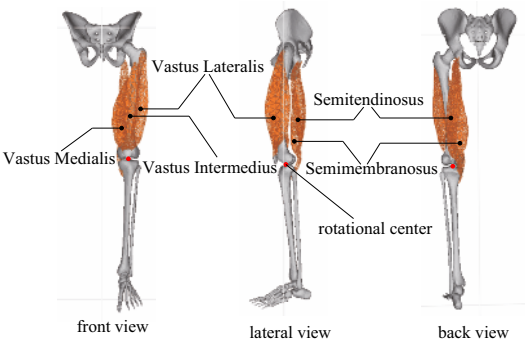


Figure 3. Knee model with 5 elastic bodies as muscles.

3 Experiments

3.1 The used model

In this paper, we made a model of left leg based on 3D human anatomical model Zygote (`www.zygote.com`). We made the two links above and beyond knee joint of left leg and connected these links with a rotational joint. Moreover, we made the finite element models of five muscles. The finite element mesh of muscles were defined based on the polygon model of Zygote. Fig.3 shows the positional relationship of muscles and bones of the model.

We decided Young's modulus and Poisson's ratio in muscle regions 3.65×10^{-2}MPa and 0.49, respectively. In regions corresponding to tendon and lig-

Table 1. Execution time in preparation of one quadratic programming.

	execution time[s]	
muscle name (number of nodes)	sequential	parallel
Vastus Medialis (1618)	2.084×10^3	2.868×10^3
Vastus Intermedius (1076)	8.346×10^2	1.272×10^3
Vastus Lateralis (1987)	3.241×10^3	4.152×10^3
Semitendinosus (2296)	4.177×10^3	5.058×10^3
Semimembranosus (1836)	2.413×10^3	3.272×10^3
All	1.275×10^4	5.058×10^3

ament, Young's modulus and Poisson's ratio were set 4.5×10^2MPa and 0.3, respectively. These physical properties were decided by reference to a physical property database (`http://cfd-duo.riken.go.jp/cbms-mp/database.php`). Also, we set the maximum isometric stress 3.43MPa.

3.2 Execution time

With the model of left leg, we compared execution time in sequential computing and that in parallel. We executed the preparation for the derivation of one quadratic problem. In the parallel computing, we used 5 threads, which is the same number of threads as muscles. The experiment was executed on the work station with a Intel(R) Core(TM) i7 CPU 880 (3.07GHz). We input 1.75×10^{-5} rad as angle of the knee joint and 1.00×10^{-4} as muscle activities of all muscles.

Table 1 shows the execution time in sequential and parallel computing. By parallel computing, we derived the quadratic problem 2.4 times faster than in sequential computing. The efficiency of parallel computing is indicated experimentally though the execution speed of each muscle declines. The total execution time in parallel was the same as that of semitendinosus, which has the largest number of nodes.This shows the computation of each muscle is executed in parallel completely.

3.3 Muscle activity estimation

With the proposed method, we estimated muscle activity from joint angle. The input of joint angle θ is $\theta = 0$rad, $\dot{\theta} = 0$rad/s, $\ddot{\theta} = 0.47124$rad/s^2 at time $t = 0.0000$s and $\theta = 6.4359 \times 10^{-5}$rad, $\dot{\theta} = 7.6577 \times 10^{-3}$rad/s, $\ddot{\theta} = 0.44768$rad/s^2 at $t = 0.01667$s. The analysis was executed with 5 threads

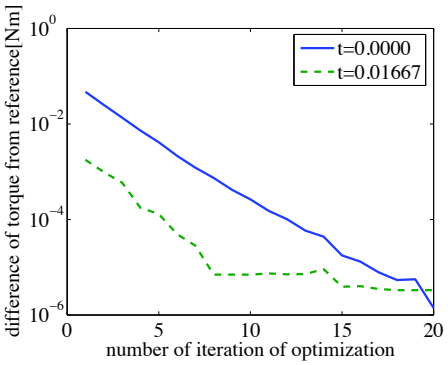


Figure 4. Convergence of torque in iterative computation. This graph shows the difference between joint torque by FEA and by inverse dynamics (reference).

on the work station with a Intel(R) Core(TM) i7 CPU 880 (3.07GHz).

The reference of joint torque calculated by inverse dynamics was 4.7124×10^{-2}Nm at $t = 0.0000$s and 4.4768×10^{-2}Nm at $t = 0.01667$s. We calculated joint torque from the obtained muscle activity of the estimation. This joint torque was 4.7123×10^{-2}Nm at $t = 0.0000$s and 4.4765×10^{-2}Nm at $t = 0.01667$s. Fig.4 presents the change of joint torque in iterative computation at each frame. The Total execution time was 24 hours and 5 minutes. As the number of iteration increased, the difference between the joint torque by FEA and that by inverse dynamics decreased. This result indicates that we could estimate muscle activity which generates the input motion data.

4 Conclusion

We proposed the method to analyze human motion considering change of force accompanied by change of muscle fiber direction with parallel programing for each muscle. We used the muscular skeletal model which consists of elastic bodies as muscles and rigid bodies as bones, and proposed the method to estimate muscle activity from time-series data of joint angle with sequential quadratic programming. Though we need to calculate joint torque and the gradient of joint torque with respect to muscle activity, these calculation can be computed in parallel for each muscle.

The proposed method was tested on the model of left leg with detailed geometry. We compared the execution time of parallel and sequential com-

putation. As a result, the parallel computing was 2.4 times faster than the sequential one. We also tested the method of muscle activity estimation. The joint torque calculated by muscle activity was converged to the required joint torque. This result indicates that we could estimate muscle activity which generates the input motion data.

The method of muscle activity estimation is independent of material models of muscles such as Saint-Kirchhoff Venant material and Mooney-Rivelin material. Therefore we can test various material models for the accurate analysis based on physiological knowledge, and this is a future work. Implementation of contact force is another future work and we can expect more precise analysis especially about the motion caused by overlapped muscles such as the motion around the trunk.

Bibliography

Human mechanical simulation: Physical property database (Japanese). `http://cfd-duo.riken.go.jp/cbms-mp/database.php`.

Zygote — 3D Human Anatomy for Animation, Illustration, CAD and Software Development. `www.zygote.com`.

A. V. Hill. The heat of shorteneing and the dynamic constants of muscle. *Proceedings of the Royal Society of London B*, 56:19–41, 1938.

F. Bachtar, X. Chen, and T. Hisada. Finite element contact analysis of the hip joint. *Medical and Biological Engineering and Computing*, 44(8): 643–651, 2006.

J. L. van Leeuwen. Optimum power output and structural design of sarcomeres. *Journal of Theoretical Biology*, 149:229–256, 1991.

K. Yamane, Y. Fujita and Y. Nakamura. Estimation of physically and physiologically valid somatosensory information. *IEEE International Conference on Robotics and Automation*, pages 2624–2630, 2005.

O. C. Zienkiewicz, R. L. Taylor and J. Z. Zhu. *The Finite Element Method: Its Basis and Fundamentals, Sixth Edition*. Butterworth-Heinemann, 2005.

S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman and D. G. Thelen. Opensim: Open-source software to create analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, 54(11):1940–1950, 2007.

T. Johansson, P. Meier and R. Blickhan. A finite-element model for mechanical analysis of skeletal muscles. *Journal of Theoretical Biology*, 206 (1):131–149, 2000.

Y. Nakamura, K. Yamane and I. Suzuki. Somatosensory computation for man-machine interface from motion-capture data and musculoskeletal human model. *IEEE Transactions on Robotics*, 21(1):99–105, 2005.

Healthy Subject Testing with the Robotic Gait Rehabilitation (RGR) Trainer

Maciej Pietrusinski * Iahn Cajigas † Paolo Bonato † and Constantinos Mavroidis *

* Northeastern University
† Harvard Medical School

Abstract The Robotic Gait Rehabilitation (RGR) Trainer has been designed to address secondary gait deviations in stroke survivors undergoing rehabilitation. In this paper we describe the operating principle of the RGR Trainer and the system′s ability to record the pelvic obliquity patterns (during normal gait and during hip hiking simulated by healthy subjects). Furthermore, we present results of experiment designed to teach new gait pattern in healthy subjects.

1 Introduction

Stroke is a leading cause of disability. Each year, 800,000 people suffer a stroke in the United States alone (Roger et al. 2011). In general, stroke survivors experience weakness and difficulties moving one side of the body (i.e. they are affected by hemiparesis), with a negative effect on the performance of motor activities such as walking. Walking allows individuals to perform activities of daily living (Guralnik et al. 1993), (Guralnik et al. 1995), and the ability to walk is strongly correlated with quality of life (Chiu et al. 2000). Hemiparesis and abnormal synergy patterns are characteristic of gait disorders following stroke. Abnormal synergy patterns include

Manuscript received April 6th, 2012. This work was supported in part by the U.S. National Science Foundation under Grant 0803622. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation. M. Pietrusinski, and C. Mavroidis are with the Department of Mechanical and Industrial Engineering, Northeastern University, Boston MA 02115 USA (phone: 617-373-4121; fax: 617-373-2921; e-mail: mavro@coe.neu.edu). I. Cajigas and P. Bonato are with the Department of Physical Medicine and Rehabilitation, Harvard Medical School and Spaulding Rehabilitation Hospital, Boston, MA, 02114, USA (phone: 617-573-2745; fax: 617-573-2769; e-mail: pbonato@partners.org).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_42,

equinus synergy, paretic synergy and reflex coactivation (Knutsson and Richards 2000). Comfortable walking speed is reduced in stroke survivors, and asymmetry of stance time during gait, a common feature following stroke, often limits walking efficiency, results in instability, and causes an aesthetically sub-optimal gait pattern. Therefore, the restoration of a normal gait pattern is an important goal of post-stroke rehabilitation. Many rehabilitation approaches have been used to promote functional recovery in stroke survivors. Unfortunately, the traditional manual rehabilitation process is labor intensive, since it often relies on a one-to-one administration of therapy, i.e. clinicians working with a single patient at a time. Robotic systems for gait retraining have been recently developed to facilitate the administration of intensive therapy. Most of the existing systems focus on the correction of primary gait deviations, such as knee hyperextension during stance and stiff-legged gait (defined as limited knee flexion during swing). Secondary gait deviations are gait abnormalities that result from compensatory movements associated with a primary gait abnormality. It is often assumed that secondary deviations would no longer be observed once primary deviations were corrected. Secondary deviations often involve the control of the pelvis. For instance, stiff-legged gait is often associated with hip-hiking and/or circumduction of the limb. Hip-hiking is an exaggerated elevation of the pelvis on the affected side of the body to allow toe clearance during swing. Circumduction of the limb is marked by an exaggerated hip abduction in combination with an exaggerated rotation of the pelvis. One would wonder whether a simple system that mechanically allows translations of the pelvis while controlling pelvic obliquity would be sufficient to achieve correction of secondary gait deviations affecting pelvic movement control. In a previous publication (Pietrusinski et al. 2011) we presented preliminary evidence of the RGR Trainers ability to teach healthy subjects a new gait pattern , which was characterized by the presence of a secondary gait deviation, namely hip-hiking with the left side of the body. That test was performed at 1.8km/h, which is a representative comfortable walking speed of hip-hiking subjects post-stroke. In this paper, we present results of pelvic trajectory recording from healthy subjects simulating hip-hiking at walking speeds which are faster than those which have been employed before. We also present results of another study, which used gait kinematics as the outcome measure to investigate how healthy subjects learn new gait pattern.

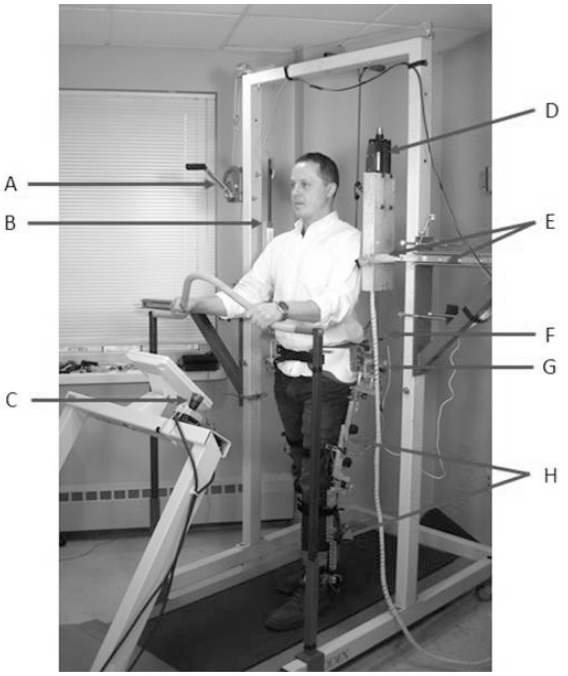


Figure 1. RGR Trainer. A − height adjust winch, B − linear potentiometer, C − emergency stop, D − linear actuator, E − linear guides, F − load cell, G − pelvic brace, H − leg brace

2 Description of RGR Trainer

Mechanical Design The RGR Trainer (Fig. 1) was designed to apply a force-field to the subject′s pelvis in order to affect its obliquity via a human-robot interface (HRI - a lower body exoskeleton) worn by the subject. The device features horizontal motion modules on the left and right sides of the subject. This design gives the subject unrestricted motion in the horizontal plane within a given range. A servo − tube linear electromagnetic actuator (model STA2508) from Copley Controls Inc. (Canton, MA, USA) is supported by one of the horizontal motion modules, allowing it to glide in the horizontal plane and apply forces in the vertical direction to control pelvic obliquity. A spherical joint is used to transfer forces from the actuator to the HRI, while a tension − compression load cell provides force feedback for control and performance evaluation purposes. The servo − tube linear ac-

tuator features integrated linear position sensing of the moving core (thrust rod), providing vertical position of the HRI on that side. On the opposite side of the body, a horizontal motion module supports a lightweight assembly with a linear potentiometer, which provides vertical position feedback of that side. A cable and winch system allows for safe adjustment of the actuation systems height to suit the user.

Device Operation The position feedback from the servo tube actuator and the linear potentiometer on the opposite side of the body are used to compute the angular position of the HRI worn by the subject in the frontal plane, as shown in Fig. 2. It is this angular position that is acted upon by a force field.

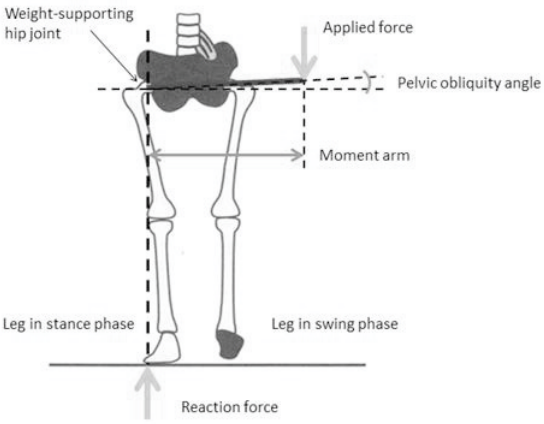


Figure 2. With one leg in swing phase (and off the ground), a moment can be applied onto the pelvis about the weight-supporting hip joint.

Control System The force field is realized in the physical sense by use of impedance − controlled linear actuator. The controller has been designed based on the method originally proposed by Hogan (Hogan 1987). The system creates a virtual spring/mass/damper attached to the HRI, such that pelvic obliquity deviations from the reference trajectory result in corrective forces applied to the pelvis. The block diagram of the impedance controller is shown in Fig. 3.

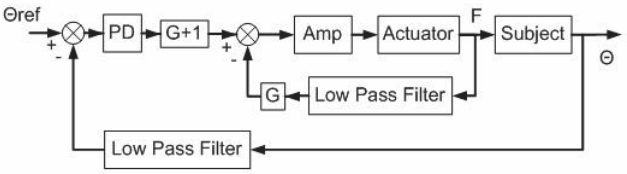


Figure 3. Impedance controller − block diagram.

Gait Estimation Algorithm The RGR Trainer applies force fields to the pelvis only when the subject′s leg on the contralateral (affected) side of the body is in swing phase, and off the ground. This requires knowledge of the subject′s location in his/her own gait cycle. A gait estimation algorithm (Aoyagi et al. 2000) was implemented in the RGR Trainer, with left and right hip and knee angular positions and velocities (8 DOFs) serving as input. This algorithm synchronizes the control system with the timing of the subect′s stepping.

3 Testing

Introduction In a previous publication we presented results from a hip-hike training experiment (Pietrusinski et al. 2011) which was performed at 1.8km/h (typical walking speed of patients post-stroke). Since then, similar experiments were performed at the healthy subjects′ self-selected comfortable walking speed (CWS) of about 3km/h. In preparation for such testing, we had collected pelvic obliquity trajectories from several healthy subjects simulating hip-hiking at their self − selected comfortable walking speed.

Hip-Hiking Pelvic Obliquity Measurement Pelvic obliquity data was collected from 8 healthy subjects who performed 50 strides under two conditions, with and without visual feedback. A mean hip-hiking trajectory across the 50 strides (with visual feedback) was found for each subject, and a mean hip-hiking trajectory across 7 of the 8 subjects was generated, pictured in Fig. 4.

Hip-Hike Training Experiment Protocol Three healthy subjects were tested in the RGR Trainer. The device was set up to minimize the perceptibility of the change in operating mode (force-field application to cause hip-hiking, followed by backdriven mode) . The intention was to trick sub-

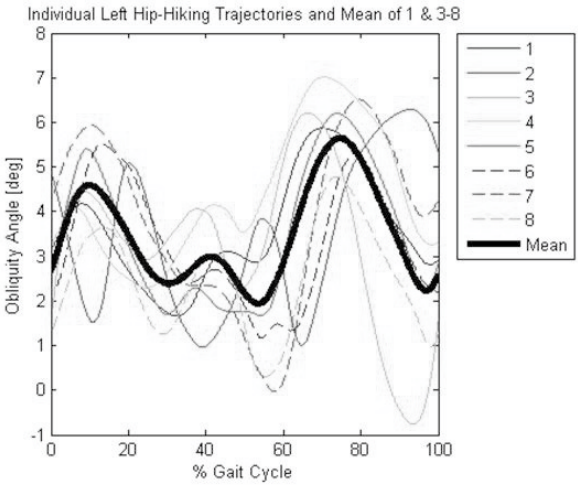


Figure 4. Inter-subject mean (7 subjects, #2 excluded) of intra-subject means (across 50 gait cycles). Gait cycle begins with left heel strike.

jects into not realizing that the RGR Trainer stopped pushing their pelvis to follow hip-hiking gait pattern. The force field was set to be activated at 55% and de-activated at 85% of the gait cycle. Each subject participated in a single session, which consisted of five trials performed continuously. Each trial used one of five force field strength magnitudes (20, 25, 30, 35 and 40N-m/deg) randomized in order. The subsequent experiment used the subjects own baseline pelvic obliquity (ref-BL) and the simulated hip-hiking pelvic obliquity (ref-HH). Each trial began with backdriven mode (Epoch 1 50 strides), followed by hip-hike training mode (Epoch 2 300 strides), then backdriven mode (Epoch 3 200 strides), followed by baseline re-training mode (Epoch 4 300 strides) and finally backdriven mode (Epoch 5 200 strides). Test session consisted of five such trials, each one with different force field strength (order randomized).

Hip Hike Training Experiment Results Subjects were instructed to follow the guidance of the RGR Trainer (assistive training). All subjects did perform the hip-hiking gait pattern under all force field strengths (sample data from one subject and one condition is shown in Fig. 5), though the magnitude of hip-hike across all subjects was less than the reference (ref-HH) by approximately 2 degrees. In all trials at the end of training epoch

the subjects immediately switched to exhibiting pelvic drop.

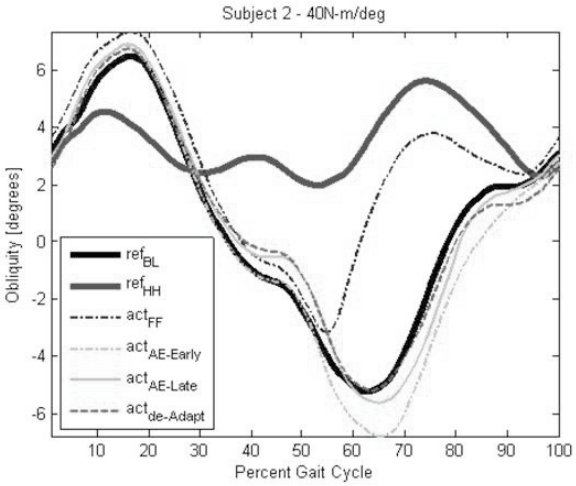


Figure 5. Subject 2 hip-hiked during the training session (act-FF), about 2 deg. below hip-hike reference (ref-HH). In epoch 3 during first 10 strides we observe increased pelvic drop (act-AE-Early) but during the last 10 cycles (act-AE-Late) he almost completely reverted to his baseline (ref-BL). Pelvic obliquity during epoch 5 is marked act-de-Adapt.

Hip-Hike Training Experiment Discussion The magnitude of the hip-hike angle exhibited by subjects during the training sessions suggests that while they complied with the force field to a certain extent, they still interacted with it via haptic feedback. Thus, upon removal of force field at the beginning of epoch 3, the subjects could sense the change in operating mode. In fact, while this was intended to be an assistive-training experiment, the subjects′ response, namely the exaggerated pelvic drop as the after-effect, is a result one would expect from resistive training.

Conclusion and Future Work The RGR Trainer and the human-robot interface have been designed to efficiently and accurately transfer force fields to the pelvis and lower body of a patient post-stroke in order to address the hip-hiking gait pattern. The RGR Trainers high backdrivability was demonstrated and used for simulated hip-hiking pelvic obliquity recording from healthy subjects. Hip-hike training experiments were performed on

three subjects, showing ability to alter subjects gait during training, and an unexpected after-effect (increased pelvic drop) was observed. This valuable result will be taken into consideration when designing subsequent experiments.

Bibliography

Aoyagi, D., Ichinose, W. E., Harkema, S. J., Reinkensmeyer, D. J., and Bobrow, J. E., *A robot and control algorithm that can synchronously assist in naturalistic motion during body-weight-supported gait training following neurologic injury*, In *IEEE Trans Neural Syst Rehabil Eng*, vol. 15, pp. 387-400, Sep 2007.

Chiu, H. C., Chern, J. Y., Shi, H. Y., Chen, S. H., and Chang, J. K., *Lower-extremity function in persons over the age of 70 years as a predictor of subsequent disability*, In *The Kaohsiung Journal of Medical Sciences*, vol. 16, pp. 285-292, 2000.

Guralnik, J. M., LaCroix, A. Z., Abbott, R. D., Berkman, L. F., Satterfield, S., Evans, D. A., and Wallace, R. B. *Lower-extremity function in persons over the age of 70 years as a predictor of subsequent disability*, In *Am J Epidemiol*, vol. 137, pp. 845-57, Apr 15 1993.

Guralnik, J. M., Ferrucci, L., Simonsick, E. M., Salive, M. E. and Wallace, R. B., *Physical functioning and health-related quality of life: before and after total hip replacement*, In *N Engl J Med*, vol. 332, pp. 556-61, Mar 2 1995.

Hogan, N., *Stable execution of contact tasks using impedance control*, In *IEEE International Conference on Robotics and Automation*, Raleigh, NC, 1987, pp. 1047-1054.

Kirker, S. G., Jenner, J. R., Simpson, D. S., and Wing, A. M., *Lower-extremity function in persons over the age of 70 years as a predictor of subsequent disability*, In *The Kaohsiung Journal of Medical Sciences*, vol. 16, pp. 285-292, 2000.

Knutsson, E. and Richards, C., *Different types of disturbed motor control in gait of hemiparetic patients*, In *Brain*, vol. 102, pp. 405-30, Jun 1979.

Pietrusinski, M., Unluhisarcikli, O., Mavroidis, C., Cajigas, I., and Bonato, P., *Design of human-Machine interface and altering of pelvic obliquity with RGR Trainer*, In *International Conference on Rehabilitation Robotics, Proceedings*, 2011.

Roger, V. L., et al., *Heart disease and stroke statistics–2011 update: a report from the American Heart Association*, In *Circulation*, vol. 123, pp. e18-e209, Feb 1 2011.

Chapter VII
Perception

The ROBOSKIN Project: Challenges and Results

Aude Billard[1], Annalisa Bonfiglio[2], Giorgio Cannata[3], Piero Cosseddu[2], Torbjorn Dahl[4], Kerstin Dautenhahn[5], Fulvio Mastrogiovanni[3], Giorgio Metta[6], Lorenzo Natale[6], Ben Robins[5], Lucia Seminara[3], Maurizio Valle[3]

[1] Ecole Polytechnique Federale de Lausanne, Switzerland
[2] University of Cagliari, Italy
[3] University of Genova, Italy
[4] University at Wales in Newport, U.K.
[5] University of Hertfordshire, U.K.
[6] Italian Institute of Technology, Italy

Abstract The goal of the ROBOSKIN project[1] is to develop and demonstrate a range of new robot capabilities based on the tactile feedback provided by a robotic skin covering large areas of the robot body. So far, a principled investigation of these issues has been limited by the lack of tactile sensing technology enabling large scale experimental activities. As a matter of fact, skin based technology and embedded tactile sensors have been mostly demonstrated only at the stage of prototypes. The new capabilities are expected to improve the ability of robots to effectively and safely operate in unconstrained environments, as well as to communicate and cooperate with each other and with humans.

1 Introduction

Tactile sensing is strategic for the design and the implementation of safe interaction processes involving robots, humans and objects present in the robot (possibly unstructured) workspace. As a matter of fact, tactile sensing provides the most important and direct feedback to control contact phenomena both in case of voluntary and non-voluntary interactions with the environment.

Beyond the classical robot interaction tasks (e.g., the *peg in hole* problem) where the onset of interaction can be either expected or planned to occur at a specific location of the robot body (typically at the end effector

[1] Please refer to the official ROBOSKIN webpage at `www.roboskin.eu`.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_43,

tip), more advanced applications require more complex forms of interaction (e.g., whole hand or whole arm grasping and manipulation or gait stability control, just to name but a few), where the location and the characteristics of the contact can not be predicted or possibly modelled in advance. Furthermore, the control of physical interaction control can play a more complex perceptive role as in *active* sensory perception control loops, for instance in the case of robot programming by demonstration paradigms or in tactile based social cognition tasks, where the physical *modes* of interaction at the same time *arise* from and originate the human-robot interaction process.

In order to tackle these new research issues both at control and perceptive levels, it is necessary to plan for a research agenda taking into account a number of aspects, namely: i) novel sensory systems have to be developed with the aim of measuring and characterizing interaction phenomena that arise over (possibly) large contact areas; ii) these sensory systems have to be properly interfaced with motor control modules to guarantee a reactive and safe interaction between the robot and the environment; iii) appropriate robot perception and cognitive strategies must be designed and implemented, which are based on the underlying sensing structure, in order to enable meaningful human-robot interaction tasks.

According to these premises, the overall objective of ROBOSKIN has been to improve the ability of robots to act efficiently and safely *during* tasks involving a large degree of human-robot interaction. To this aim the project focused on:

- the study of sensing technologies and methodologies for the development of distributed and modular components for building robot skin,
- the study of control and perception mechanisms that are required to develop cognitive mechanisms exploiting tactile feedback to improve human-robot interaction capabilities.

So far, tactile sensing has been typically assumed to be a basic perception strategy to be used when purposively manipulating objects or to properly control the effects caused by robot motion when in contact with the environment. On the contrary, ROBOSKIN aimed at addressing cognitive aspects related to tactile sensing, tactile perception and motion control from a different perspective. On the one hand, we have been interested in investigating problems related to self-awareness and cognitive development through mechanisms exploiting tactile sensing to allow the robot to construct a model of its own body. On the other hand, we aimed at studying mechanisms of perception and safe reaction when objects or other physical agents (typically, humans) interact with the robot by touch, and to prove their effectiveness within application domains where touch can play a ma-

jor role, such as robot programming by demonstration and skin based social cognition.

The article is organized as follows. Section 2 introduces the main ROBOSKIN objectives and discusses how the objectives have been met, with a specific emphasis on key achievements. Conclusions follow.

2 Results and Major Achievements

Research directions pursued in ROBOSKIN have been identified as a key framework to progress further research in physical HRI as well as sensing and control in a more general perspective. For this reason, ROBOSKIN wishes to prove that:

1. There exists a fundamental role of tactile sensing and perception for the implementation of a class of application relevant HRI tasks, which motivates the development of large scale robot skin-like systems.
2. It is possible to implement procedures and methods for the development of large scale robot skin systems, using *state of the art* technology and *accessible servicing facilities*, which can be tailored for different robot platforms.
3. It is possible to develop a SW framework bridging the tactile HW with the perceptive and the control SW modules, also supporting the implementation of high level skin-based cognitive robot interaction tasks.

According to these objectives, during the past three years the ROBOSKIN consortium set out a research agenda pursuing technological, modelling and application results, which are briefly described as follows.

Result 1. *Specification of procedures and technology for the construction of robot skin systems for laboratory level robotic experiments.* Skin design tools have been developed to aid robot skin design processes (Anghinolfi et al., 2012; Maiolino et al., 2011), see Figure 1. On the one hand, algorithms for the optimal placement (according to a number of criteria) of skin patches over robot body parts, as well as for the optimal networking and routing of skin modules have been designed and developed. On the other hand, SW tools to specifically tune geometric and mechanic parameters related to the current skin prototypes have been developed. With the aim of tailoring robot skin for different robot platforms, it is possible to select different materials (according to their mechanical properties), varying skin geometrical parameters and to define trade-offs between these parameters and other functional requirements such as *sensitivity* and *accuracy* in the response.

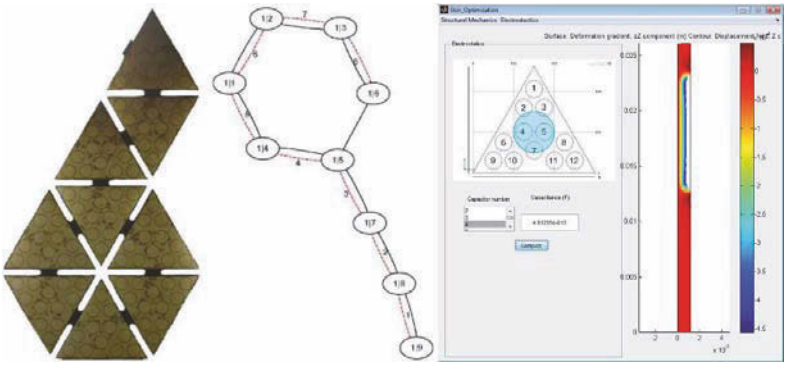


Figure 1. Left: networking design in skin patches. Right: a screenshot of the Skin Design Toolbox.

Figure 2. Robots covered with skin patches: iCub, Kaspar and NAO.

Result 2. *Implementation of large scale, multimodal, modular robot skin with embedded electronics on different robot systems.* Two different sets of activities have been carried out. On the one hand, real-time networking solutions for connecting large-scale tactile systems have been developed (Baglini et al., 2010): both general purpose and custom solutions have been investigated. On the other hand, specific procedures have been envisaged to cover robots of different shape with large-scale tactile systems (Schmitz et al., 2011), see Figure 2.

Result 3. *Implementation of middleware for tactile data integration and interpretation, including integrated sensorimotor strategies for reactive protective reflexes, and for robot tactile self-exploration.* The goal of the SW framework (Youssefi et al., 2011) is to provide an abstract and HW independent representation of the skin, which is organized in logical units

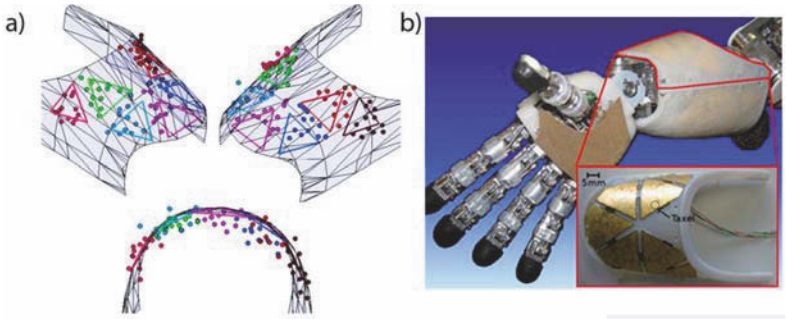


Figure 3. Left: results of the skin spatial calibration process. Right: the actual body part to be calibrated.

representing different body parts. Skin SW technologies include algorithms and data structures allowing tactile data to travel from the lowest level (i.e., the actual sensors) up to user level applications. Both general-purpose and robot-specific tactile data processing architectures have been investigated and experimentally evaluated in terms of real-time performance, specifically taking into account bandwidth, jitter and reliability issues. Specific emphasis has been put on the so-called *skin spatial calibration problem*, i.e., the problem of self-estimating the location of tactile elements mounted on a robot body part, see Figure 3. To this aim, different solutions have been investigated (Cannata et al., 2010a; Prete et al., 2011; McGregor et al., 2011).

Result 4. *Implementation of an architecture for touch-based social interaction, and development of classification algorithms for touch-based social interaction.* Robot skin makes the design and implementation of a number of robot behaviours possible both at the reflexive and purposive levels. On the one hand, skin based protective reflexes have been developed, which are based on *reflex receptive fields* as reported in studies from human subjects (Pierris and Dahl, 2010). On the other hand, methods to obtain tactile based robot motion behaviours have been investigated (Cannata et al., 2010b,c).

Result 5. *Design an autonomous robot capable of skin-based interaction specifically for children with autism.* Aspects related to cognitive and embodied learning have been investigated, with a specific emphasis on tactile social interaction tasks with children with autism. The teleoperated robot

Figure 4. Children interacting with the robot Kaspar.

Kaspar (see Figure 4) has been used as a mediator to investigate general cognitive learning with very low functioning children with autism, specifically exploring *sad and happy* expressions, *cause and effect* relationships as well to develop coordination mechanisms (Wainer et al., 2010).

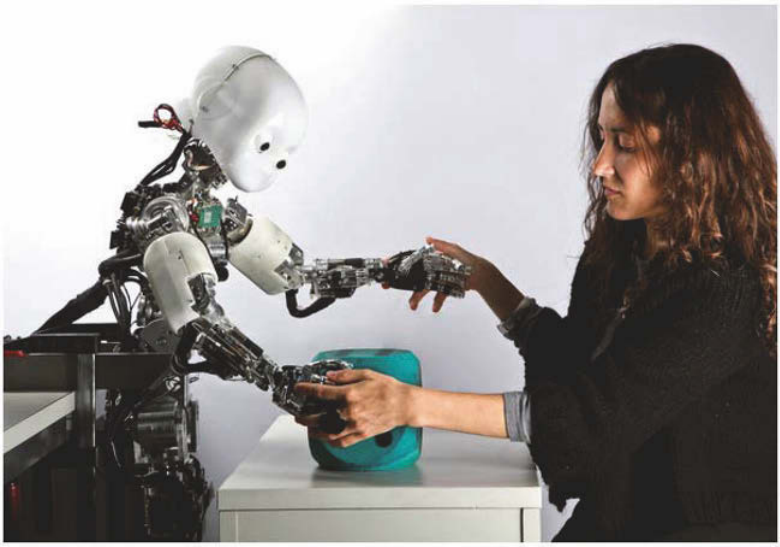


Figure 5. Tactile correction are provided to the robot after the demonstration phase.

Result 6. *Implementation of new means of teaching skills to a robot, based on force-control, through touch-based kinesthetic training.* The main idea is that tactile feedback naturally extends the idea of teaching robots as hu-

mans teach other humans (see Figure 5). The somewhat classical paradigm of robot programming by demonstration has been extended with correction policies based on tactile feedback. In particular, after the off-line demonstration phase has occurred, on-line feedback is provided through the skin mounted on relevant robot body parts (Sauser et al., 2012).

3 Conclusions

In order to measure the success of ROBOSKIN and to assess the major results described in Section 2, a number of demonstrations have been set up that are strictly related to the main project objectives. However, the main results consist in the fact that four different robot platforms have been provided with a skin system based on the technology developed in the context of ROBOSKIN, namely iCub, NAO (by Aldebaran Robotics), Kaspar and a modular robot arm from Schunk GmbH. This demonstrates the effectiveness of the developed HW and SW technology to adapt to different robot shapes, control architectures and tasks. The overall results of ROBOSKIN pave the way for further advances in tactile sensing and touch-based human-robot interaction processes.

4 Acknowledgements

The research leading to these results has received funding from the European Commission's Seventh Framework Programme (FP7) under Grant Agreement no. 231500 (ROBOSKIN). The authors would like to acknowledge all the people involved in the project in the past three years.

Bibliography

D. Anghinolfi, G. Cannata, F. Mastrogiovanni, C. Nattero, and M. Paolucci. Heuristic approaches for the optimal wiring in large scale robotic skin design. *Computers and Operation Research*, 39(11):2715–2724, 2012.

E. Baglini, G. Cannata, and F. Mastrogiovanni. Design of an embedded networking infrastructure for whole-body tactile sensing in humanoid robots. In *Proceedings of the 2010 IEEE-RAS International Conference on Humanoid Robotics (HUMANOIDS 2010)*, Nashville, USA, December 2010.

G. Cannata, S. Denei, and F. Mastrogiovanni. Towards automated self-calibration of robot skin. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*, Anchorage, USA, May 2010a.

G. Cannata, S. Denei, and F. Mastrogiovanni. A framework for representing interaction tasks based on tactile data. In *Proceedings of the 2010 IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2010)*, Viareggio, Italy, September 2010b.

G. Cannata, S. Denei, and F. Mastrogiovanni. Tactile sensing: Steps to artificial somatosensory maps. In *Proceedings of the 2010 IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2010)*, Viareggio, Italy, September 2010c.

P. Maiolino, T-H-L. Lee, A. Schmitz, F. Mastrogiovanni, and G. Cannata. A toolbox for supporting the design of large-scale tactile systems. In *Proceedings of the 2011 IEEE-RAS International Conference on Humanoid Robotics (HUMANOIDS 2011)*, Bled, Slovenia, November 2011.

S. McGregor, D. Polani, and K. Dautenhahn. Generation of tactile maps for artificial skin. *PLoS ONE*, 11(6), 2011.

G. Pierris and T. S. Dahl. Compressed sparse code hierarchical som on learning and reproducing gestures in humanoid robots. In *Proceedings of the 2010 IEEE International Symposium in Robot and Human Interactive Communication (ROMAN 2010)*, Viareggio, Italy, September 2010.

A. Del Prete, S. Denei, L. Natale, F. Mastrogiovanni, F. Nori, G. Cannata, and G. Metta. Skin spatial calibration using force / torque measurements. In *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, San Francisco, CA, USA, October 2011.

E. Sauser, B. Argall, G. Metta, and A. Billard. Iterative learning of grasp adaptation through human corrections. *Robotics and Autonomous Systems*, 60(1):55–71, 2012.

A. Schmitz, P. Maiolino, M. Maggiali, L. Natale, G. Cannata, and G. Metta. Methods and technologies for the implementation of large-scale robot tactile sensors. *IEEE Transactions on Robotics*, 3(27):389–400, 2011.

J. Wainer, K. Dautenhahn, B. Robins, and F. Amirabdollahian. Collaborating with kaspar: Using an autonomous humanoid robot to foster cooperative dyadic play among children with autism. In *Proceedings of the 2010 IEEE-RAS International Conference on Humanoid Robotics (HUMANOIDS 2010)*, Nashville, USA, December 2010.

S. Youssefi, S. Denei, F. Mastrogiovanni, and G. Cannata. A middleware for whole body skin-like tactile systems. In *Proceedings of the 2011 IEEE-RAS International Conference on Humanoid Robotics (HUMANOIDS 2011)*, Bled, Slovenia, November 2011.

Modeling Verticality Estimation During Locomotion

Ildar Farkhatdinov [1] Hannah Michalska [2]
Alain Berthoz [3] Vincent Hayward [1]

[1] UPMC Univ Paris 06, UMR 7222,
Institut des Systèmes Intelligents et de Robotique, Paris, France
[2] Department of Electrical and Computer Engineering,
McGill University, Montréal, Qc, Canada
[3] Laboratoire de Physiologie de la Perception et de l'Action,
Collège de France, Paris, France

Abstract Estimation of the gravitational vertical is a fundamental problem faced by locomoting biological systems and robots alike. A robotic model of a vestibular system is suggested with the purpose of explaining an observed phenomenon—head stabilization during locomotion. The mechanical model of the vestibular system comprises a damped inclinometer and an inertial measurement unit which are mounted on an actuated orienting platform (a robotic head). Generic linear control is employed to stabilize the head-platform while the vestibular system exercises an extended Kalman filter algorithm to estimate the gravitational direction in space. It is demonstrated that stabilization of the head-platform is essential in achieving accurate verticality estimation as it attenuates the disturbances generated by locomotion and simplifies state observation in a non-inertial frame, without the need for fixed external beacons.

1 Introduction

Mobile robotic systems including humanoids, mobile robots, or drones typically employ accelerometers and gyrometers packs (known as 'inertial measurement units' or IMU) to obtain information useful for navigation, orientation, balance, attitude control, and other needs, independently from references to external, fixed landmarks [2].

Most humanoids have an IMU attached to their trunk, near the center of mass, to provide balance information [4]. In humans and animals, however, the vestibular system is head-located. This organ is known to serve many functions [1], including gaze stabilization, spatial orientation, and others.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_44,

It has been proposed by Pozzo and Berthoz [8, 9] that, through head stabilization in orientation, the inertial measurements provided by the vestibular system facilitate the creation of a ground-independent, quasi-inertial, mobile frame from which postural control can be more effectively performed.

With the purpose of clarifying this observation further, a nonlinear mechanical model for gravitational vertical estimation is introduced including an inclinometer combined with an IMU, as proposed in [4]. Here, a more general model of such a robotic vestibular system is presented and the nonlinear gravitational verticality estimation problem is solved using an extended Kalman filter (EKF).

2 Robotic and Human Verticality Estimation

Accelerometers typically report noisy measurements because they pick-up vibrations added to the low frequency components of the signal. Gyroscopic measurements also suffer from bias and are highly sensitive to dynamic errors. To combat these problems, state observers and sensor fusion methods have been proposed [2]. In reference [5], a Kalman filter was used to estimate the vertical direction from tilt measurements in the linearized, planar case. Kalman filters were also applied to attitude estimation of accelerated rigid bodies in three dimensions from fused measurements of gyroscopes and accelerometers [10]. In reference [7], a nonlinear observer for attitude estimation based on gyroscope measurements is described, but only kinematic relationships are considered. A similar problem is solved in [3], where a nonlinear observer combined inertial and visual information is employed. None of these works consider the full system dynamics.

Studies have shown that humans spontaneously stabilize their heads during various locomotion tasks: free walking, walking in place, running in place and hopping [8, 9]. This behavior may reflect the existence of a synergistic interaction between the measurement of rotations by the semi-circular canals and the measurement of translations by the utriculus and sacculus organs. The plane of head stabilization generally is determined by the task and is linked to the gaze direction. At the neural level, it is believed that the estimation of the vertical direction from various sensor inputs is a fundamental brain function and their neural correlates have been studied in humans and animals models [1].

It is an enticing idea to replicate the head stabilization behavior in robots, which is investigated here. Besides suggesting improved robotic design, such replication may provide new insights in the brain functions related to vestibular inputs.

3 Model

Key aspects of the function of the vestibular system can be emulated by a liquid-based inclinometer (Model A900 from Applied Geomechanics). The sensing element is a glass vial partially filled with a conductive liquid, see Figure 1, left panel. When the sensor is level, the four internal electrodes are immersed in the liquid at equal depths; otherwise, the depth of immersion of the electrodes changes, altering the electrical resistance between matched pairs of electrodes. The sensor is mounted on a oriented platform modeled here as a gimble mechanism with two actuators, M1 and M2. A damped pendulum in space with concentrated mass m, viscous damping β and length l, see Fig. 1 right panel, can be used as a mechanical model of the sensor.

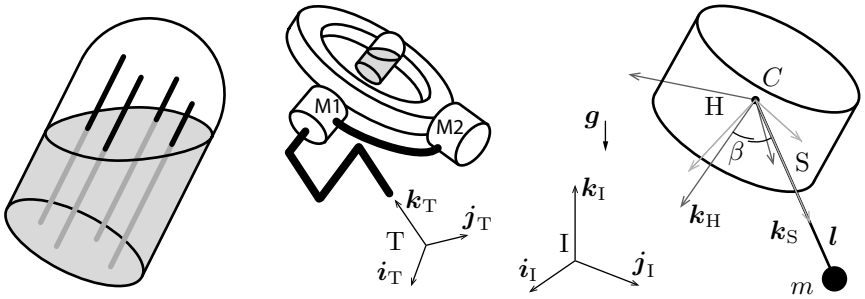


Figure 1. Inclinometer. Left: Liquid partially-filled vial with electrodes. Verticality measurements are affected by fictitious forces when the frame is not Galilean. Middle: actuated gimble mechanism as a model for a head articulated with respect to an arbitrarily moving trunk located by frame T. Frame I is the inertial frame. The platform also supports an IMU (not represented) located near its center. Right: Frame H, in medium gray, is attached to the head and has its origin at C, the center of mass of the platform. Frame S, in light gray, is attached to the pendulum modeled as concentrated mass m located at the end of vector ${}^{\mathrm{S}}\boldsymbol{l} = [0\ 0\ l]^\top$, aligned with unit vector $\boldsymbol{k}_\mathrm{S}$, having its origin in C, and $\boldsymbol{g}$ is the acceleration of gravity.

In the foregoing, vector and tensor quantities that are sensitive to the frame in which they are expressed receive a left superscript to indicate it. Thus, a rotation transformation written as ${}^{\mathrm{B}}\mathbf{R}_\mathrm{A}$ is a rotation matrix that transforms, by left multiplication, vectors expressed in frame A into vectors expressed in some frame B. Alternatively, it expresses the orientation of B with respect to A. If B is a frame moving with respect to A and ${}^{\mathrm{A}}\boldsymbol{q}$ is a position vector expressed in frame A, then $\mathrm{d}^{\mathrm{B}}\boldsymbol{q}/\mathrm{d}t = \mathrm{d}^{\mathrm{A}}\boldsymbol{q}/\mathrm{d}t + {}^{\mathrm{B}}\boldsymbol{\omega}_\mathrm{A} \times {}^{\mathrm{A}}\boldsymbol{q}$, where

${}^{\mathrm{B}}\boldsymbol{\omega}_{\mathrm{A}}$ is the angular velocity of frame A relatively to B, expressed in frame B. Given a vector $\boldsymbol{q}$, the symbol $[\boldsymbol{q}]_{\times}$ is used to denote the corresponding skew-symmetric matrix, facilitating differentiation.

The angular momentum of a body, $\boldsymbol{h}$, is the product of a tensor of inertia, $\mathbf{J}$, and an angular velocity, $\boldsymbol{\omega}$, where the angular velocity and the tensor are expressed in the same frame. If the tensor is expressed in a body-fixed frame, the formulae become much simpler since the tensor is constant.

The model requires the definition of four frames: (i) frame I is the inertial frame with unit vectors, $\{{}^{\mathrm{I}}\boldsymbol{i}_{\mathrm{I}}, {}^{\mathrm{I}}\boldsymbol{j}_{\mathrm{I}}, {}^{\mathrm{I}}\boldsymbol{k}_{\mathrm{I}}\}$; (ii) frame H is the body-fixed head frame with unit vectors, $\{{}^{\mathrm{I}}\boldsymbol{i}_{\mathrm{H}}, {}^{\mathrm{I}}\boldsymbol{j}_{\mathrm{H}}, {}^{\mathrm{I}}\boldsymbol{k}_{\mathrm{H}}\}$ and with origin at point, C, coinciding with the center of mass of the head; (iii) frame S is the body-fixed pendulum coordinate frame with unit vectors, $\{{}^{\mathrm{I}}\boldsymbol{i}_{\mathrm{S}}, {}^{\mathrm{I}}\boldsymbol{j}_{\mathrm{S}}, {}^{\mathrm{I}}\boldsymbol{k}_{\mathrm{S}}\}$ such that its $\boldsymbol{k}_{\mathrm{S}}$ axis is aligned with the arm of the pendulum and the pivot coincides with C; (iv) finally, frame T is the frame attached to the trunk of the robot to which the head is attached and that can move in arbitrary ways.

A spherical pendulum is attached to the head which is translated with acceleration ${}^{\mathrm{I}}\boldsymbol{a}$. A torque, ${}^{\mathrm{H}}\boldsymbol{\tau}$, is applied to the head platform by the actuated gimbal. The time derivatives of the angular momenta of the pendulum and of the head are,

$$\dot{\boldsymbol{h}}_{\mathrm{S}} = \frac{\mathrm{d}}{\mathrm{d}t}\left({}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}}\right) = {}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}}\,{}^{\mathrm{I}}\dot{\boldsymbol{\omega}}_{\mathrm{S}} + {}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}} \times {}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}},$$

$$\text{and} \qquad \dot{\boldsymbol{h}}_{\mathrm{H}} = \frac{\mathrm{d}}{\mathrm{d}t}\left({}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}\right) = {}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}}\,{}^{\mathrm{I}}\dot{\boldsymbol{\omega}}_{\mathrm{H}} + {}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}} \times {}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}.$$

The influence of the pendulum on the motion of the head is disregarded, as the mass of the pendulum is considered negligibly small, however, the viscous torque resulting from the difference of the angular velocities of the two bodies must be accounted for. Under these assumptions, the sum of all the moments acting on the pendulum-head system are,

$$\begin{aligned} {}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}}\,{}^{\mathrm{I}}\dot{\boldsymbol{\omega}}_{\mathrm{S}} &= -\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}} \times {}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}} + mg\,{}^{\mathrm{S}}\boldsymbol{l} \times {}^{\mathrm{S}}\mathbf{R}_{\mathrm{I}}\,\boldsymbol{k}_{\mathrm{I}} \\ &\quad - m\,{}^{\mathrm{S}}\boldsymbol{l} \times {}^{\mathrm{S}}\mathbf{R}_{\mathrm{I}}\,{}^{\mathrm{I}}\boldsymbol{a} - \beta\left({}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}} - {}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}\right), \\ {}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}}\,{}^{\mathrm{I}}\dot{\boldsymbol{\omega}}_{\mathrm{H}} &= -\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}} \times {}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}}\,{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}} + {}^{\mathrm{H}}\boldsymbol{\tau}. \end{aligned} \tag{1}$$

where ${}^{\mathrm{S}}\mathbf{R}_{\mathrm{I}}\,{}^{\mathrm{I}}\boldsymbol{a}$ is the measurement returned by the IMU's accelerometer. The rates of change of the rotation matrices obey the kinematic relationships,

$$\frac{\mathrm{d}\,{}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}}{\mathrm{d}t} = \left[{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}}\right]_{\times}{}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}, \quad \frac{\mathrm{d}\,{}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}}{\mathrm{d}t} = \left[{}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}\right]_{\times}{}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}. \tag{2}$$

The inclinometer readings correspond to two angles which describe the orientation of the liquid surface, denoted by a normal vector with respect to

its base. These angles can be expressed through cross products of the corresponding unit vectors of the inclinometer's and head's frame. The outputs of the system comprise the inclinometer readings,

$$\begin{aligned} y_1 &= \arcsin \left\| {}^{\mathrm{I}}\boldsymbol{j}_{\mathrm{H}} \times {}^{\mathrm{I}}\boldsymbol{j}_{\mathrm{S}} \right\| = \arcsin \left\| {}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}\, {}^{\mathrm{H}}\boldsymbol{j}_{\mathrm{H}} \times {}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}\, {}^{\mathrm{S}}\boldsymbol{j}_{\mathrm{S}} \right\|, \\ y_2 &= \arcsin \left\| {}^{\mathrm{I}}\boldsymbol{i}_{\mathrm{H}} \times {}^{\mathrm{I}}\boldsymbol{i}_{\mathrm{S}} \right\| = \arcsin \left\| {}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}\, {}^{\mathrm{H}}\boldsymbol{i}_{\mathrm{H}} \times {}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}\, {}^{\mathrm{S}}\boldsymbol{i}_{\mathrm{S}} \right\|. \end{aligned} \tag{3}$$

Note that that in steady state inclinometer readings correspond to the head's orientation with respect to the vertical: pitch angle, Θ_x, and roll angle, Θ_y. The IMU's rate gyros measurements are also included in the system's output because they relate directly to the sought unknown states, namely, the head orientation with respect to gravity,

$$\{y_3, y_4, y_5\} = \text{components} \left({}^{\mathrm{H}}\mathbf{R}_{\mathrm{I}}\, {}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}} \right), \tag{4}$$

which completes the model by collecting (1)–(4). The system states are the elements of the matrices ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}$ and ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}$ and of the vectors ${}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}}$ and ${}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}$.

A crude decentralized 'joint space' control, see [6], is employed to find the motor torques, $\boldsymbol{m} = [m_1, m_2]^{\mathrm{T}}$, that stabilize the head in the horizontal place given an estimate of ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}}$. The map, ${}^{\mathrm{T}}\mathbf{R}_{\mathrm{H}} = \Lambda(\theta_1, \theta_2)$, relates the motor shaft angles, $\boldsymbol{\theta} = [\theta_1, \theta_2]^{\mathrm{T}}$, to the orientation of the head with respect to the trunk. The head is horizontal when ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}} = {}^{\mathrm{I}}\mathbf{R}(\boldsymbol{k}_{\mathrm{I}})$ where ${}^{\mathrm{I}}\mathbf{R}(\boldsymbol{k})$ represents the rotation about vector $\boldsymbol{k}_{\mathrm{I}}$ that cannot be effected. The orientation ${}^{\mathrm{T}}\mathbf{R}_{\mathrm{H}}$ can be known from the joint angles. If $\theta_1, \theta_2 < \pi/4$, then Λ is invertible and we compute the desired joint angles by solving $\boldsymbol{\theta}_d = \Lambda^{-1}({}^{\mathrm{I}}\mathbf{R}(-\boldsymbol{k}_{\mathrm{I}}){}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}})$. When the servo error is zero, the head orientation, ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}} = {}^{\mathrm{I}}\mathbf{R}_{\mathrm{T}}\, {}^{\mathrm{T}}\mathbf{R}_{\mathrm{H}}$, becomes ${}^{\mathrm{I}}\mathbf{R}_{\mathrm{T}}{}^{\mathrm{I}}\mathbf{R}(-\boldsymbol{k}_{\mathrm{I}}){}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}} \Rightarrow {}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}} = {}^{\mathrm{I}}\mathbf{R}(\boldsymbol{k}_{\mathrm{I}})$. The motor torques are obtained from the outputs of proportional-derivative controllers $\boldsymbol{m} = \mathrm{PD}(\boldsymbol{\theta}_d - \boldsymbol{\theta})$ that achieve regulation of $\boldsymbol{\theta}$ to the desired value $\boldsymbol{\theta}_d$. If $\mathbf{M}$ denotes the Jacobian matrix of Λ, ${}^{\mathrm{H}}\boldsymbol{\tau} = \mathbf{M}^{-\mathrm{T}}\boldsymbol{m}$ completes the 'joint space' control action.

4 Extended Kalman Filter

Define a state vector $\boldsymbol{x} = \left({}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{S}}\, {}^{\mathrm{I}}\mathbf{R}_{\mathrm{S}}\, {}^{\mathrm{I}}\boldsymbol{\omega}_{\mathrm{H}}\, {}^{\mathrm{I}}\mathbf{R}_{\mathrm{H}} \right)^{\mathrm{T}}$ made of the elements of the vectors where the matrices are arranged in a single vector. The system is then conveniently expressed in the form,

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{w}_{k-1}), \quad \boldsymbol{y}_k = h(\boldsymbol{x}_k, \boldsymbol{\nu}_k),$$

where $\boldsymbol{u} = \left({}^{\mathrm{I}}\boldsymbol{a}\, {}^{\mathrm{H}}\boldsymbol{\tau} \right)^{\mathrm{T}}$ is an input due to the movement of the robot combined with the torque applied to the head. The random variables $\boldsymbol{w}_k$ and $\boldsymbol{\nu}_k$ represent the process and measurement noise. The function f that relates

the state at step $k-1$ to the state at time step k comprises (1) and (2). The output function, h, relates the state $\boldsymbol{x}_k$ to the measurement $\boldsymbol{y}_k$ comprises (3) and (4). Linearizing around an estimate gives,

$$\boldsymbol{x}_k \approx \tilde{\boldsymbol{x}}_k + A(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}) + W\boldsymbol{w}_{k-1}, \boldsymbol{y}_k \approx \tilde{\boldsymbol{y}}_k + H(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}) + V\boldsymbol{\nu}_k.$$

where $\boldsymbol{x}_k$ are the actual state, $\tilde{\boldsymbol{x}}_k$ its estimate, $\boldsymbol{y}_k$ the measurement, $\tilde{\boldsymbol{y}}_k$ its estimate, $\hat{\boldsymbol{x}}_k$ the a posteriori state estimate. The matrices are,

$$A = \frac{\partial f}{\partial \boldsymbol{x}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{0}), \qquad W = \frac{\partial f}{\partial \boldsymbol{w}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{0}),$$
$$H = \frac{\partial h}{\partial \boldsymbol{x}}(\tilde{\boldsymbol{x}}_k, \boldsymbol{0}), \qquad V = \frac{\partial h}{\partial \nu}(\tilde{\boldsymbol{x}}_k, \boldsymbol{0}).$$

The state update equations are

$$\hat{\boldsymbol{x}}_k^- = f(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k-1}, 0), \qquad P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T,$$
$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}, \quad \hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + K_k(\boldsymbol{y}_k - h(\hat{\boldsymbol{x}}_k^-, 0)),$$
$$P_k = (I - K_k H_k) P_k^-.$$

5 Simulation Results

The parameters were given reasonable values, $m = 50$ g, $l = 0.06$ m, ${}^{\mathrm{S}}\mathbf{J}_{\mathrm{S}} = m\,\mathrm{diag}[l^2, l^2, \frac{1}{20}l^2]$, ${}^{\mathrm{H}}\mathbf{J}_{\mathrm{H}} = \mathrm{diag}[0.125\ 0.125\ 0.125]$ kg·m^2, $\beta = 0.001$ N·ms. The controller was tuned for a critically damped response, with $k_p = 32$ N·m/rad and $K_d = 5$ N·m·s/rad. The sampling period was 1 ms. Estimation was updated every 20 ms. The initial conditions for the EKF differed from that of the system by 0.1 rad. The standard deviation of the process noise, $\boldsymbol{w}_k$, was set to ± 0.001 and that of the measurement noise, $\boldsymbol{v}_k$, to ± 0.02. Moreover, to test robusteness, the model parameters known to the EKF were assumed to differ substancially from the true values, $\tilde{m} = 60$ g, $\tilde{l} = 0.04$ m, ${}^{\mathrm{H}}\tilde{\mathbf{J}}_{\mathrm{H}} = \mathrm{diag}[0.15\ 0.15\ 0.15]$, $\tilde{\beta} = 0.0005$ N·ms.

The tests involved a scenario, `A`, where the head underwent an oscillatory movement which resulted in a three-dimensional trajectory, Fig. 2a. A sudden impact at time 5 s which led to an 30 m/s^2 acceleration spike after which the robot was stopped. In a first condition, the head was rigidly attached to the trunk, that is, ${}^{\mathrm{T}}\mathbf{R}_{\mathrm{H}} = \mathbf{I}$ at all times. In a second the head was stabilized in the horizontal plane using the control outlined earlier. The estimation errors were much smaller when the head was stabilized although the measurement was similarly perturbed.

In another scenario, `B`, the robot jumped in the x-direction as in Fig. 2d. At 'take off' the robot was accelerated in the upward z-direction and at

'landing' the robot experienced an impact from the ground, after which the robot was stopped, Fig. 2d) and (Fig. 2e). The estimation errors were also smaller for the case of horizontally stabilized head. For clarity, the results are presented in Fig. 2 with the noise removed, but noise was present during the simulations.

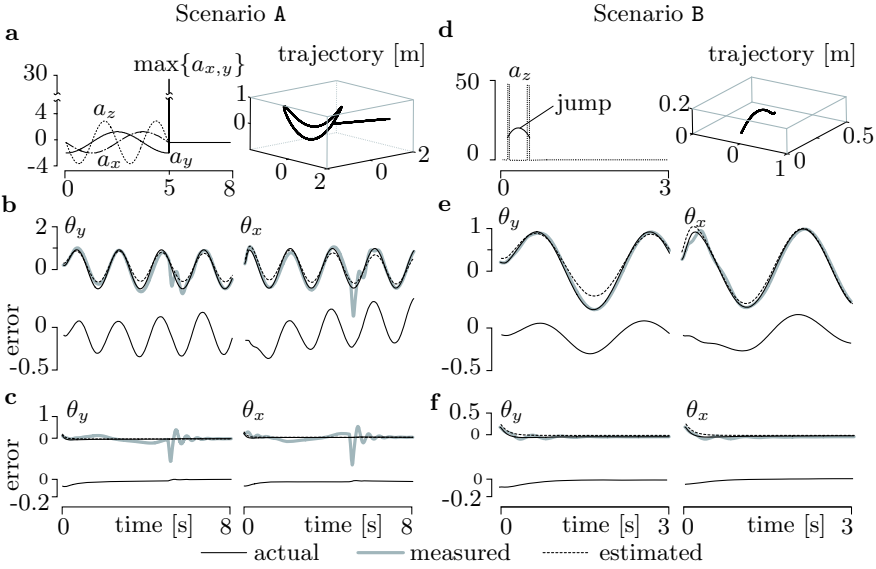


Figure 2. Simulation results. **a**, periodic trajectory and sudden impact at $t = 5\ s$. **b**, non-stabilized head orientation. **c**, stabilized head orientation. **d**, jumping trajectory with two acceleration spikes. **e**, non-stabilized head orientation. **f**, stabilized head orientation. Angles θ_x and θ_y denote the head orientation around $\boldsymbol{k}_\mathrm{I}$ and $\boldsymbol{k}_\mathrm{I}$ respectively.

6 Conclusion and Discussion

Locating the vestibular system in the head has a number of combined advantages, including providing a flexible platform that assists seeking and tracking targets with vision and audition. Here, we developed a dynamic model of what could be called a "robotic vestibular system" comprising a damped inclinometer and an IMU. With this model, we showed that locating the vestibular system in the head enabled inertial stabilization with the consequence of a reduction of the number states to be estimated, while pro-

viding a quasi-inertial reference frame that can facilitate control. We also found that stabilization was especially helpful in the face of uncertainty in the system model. We would like to argue that head stabilization would be of benefit to any robot operating in a non-inertial frame and that such feature should be replicated in their designs.

Bibliography

[1] D.E. Angelaki and K. E. Cullen. Vestibular system: the many facets of a multimodal sense. *Annual Reviews of Neuroscience*, 31:125—150, 2008.

[2] B. Barshan and H. F. Durrant-Whyte. Inertial navigation systems for mobile robots. *IEEE T. on Robotics and Automation*, 11(3):328 –342, 1995.

[3] S. Bras, R. Cunha, J. F. Vasconcelos, C. Silvestre, and P. Oliveira. A nonlinear attitude observer based on active vision and inertial measurements. *IEEE T. on Robotics*, 27(4):664–677, 2011.

[4] I. Farkhatdinov, V. Hayward, and A. Berthoz. On the benefits of head stabilization with a view to control balance and locomotion in humanoids. In *Proc. of the 11th IEEE-RAS Int. Conf. on Humanoid Robots*, pages 147—152, 2011.

[5] J. Leavitt, A. Sideris, and J. E. Bobrow. High bandwidth tilt measurement using low-cost sensors. *IEEE/ASME T. on Mechatronics*, 11(3): 320–327, 2996.

[6] R. P. Paul. *Robot manipulators: mathematics, programming, and control*. MIT Press, 1981.

[7] J. M. Pflimlin, T. Hamel, and P. Souères. Nonlinear attitude and gyroscope's bias estimation for a VTOL UAV. *Int. J. of Systems Science*, 38(3):197–210, 2007.

[8] T. Pozzo, A. Berthoz, and L. Lefort. Head stabilisation during various locomotor tasks in humans. *Experimental Brain Research*, 82(1):97–106, 1990.

[9] T. Pozzo, Y. Levik, and A. Berthoz. Head and trunk movements in the frontal plane during complex dynamic equilibrium tasks in humans. *Experimental Brain Research*, 106(2):327–338, 1995.

[10] H. Rehbinder and X. Hu. Drift-free attitude estimation for accelerated rigid bodies. *Automatica*, 40(4):653—659, 2004.

The Whole-Arm Exploration of Harsh Environments

Francesco Mazzini † and Steven Dubowsky †

† The Field and Space Robotics Laboratory, Massachusetts Institute of Technology, Cambridge MA 02139

Abstract This work develops a method for tactilely mapping unknown harsh environments such as oil wells, using a manipulator equipped with only position sensors. Because contact with the environment may occur anywhere on the manipulator, determining the contact location is challenging. Here a method is developed, based on a probabilistic classification of the data according to the contact location on the manipulator, and the reconstruction of the surface using such classified data. The approach effectiveness is demonstrated in several case studies and laboratory experiments.

1 Introduction

Here, the robotic tactile exploration of very harsh environments such as those found in pipes, mines, sewers, nuclear facilities and oil wells is considered, when the environment conditions prevent the use of range, force or tactile sensors. The exploration of oil wells is often required to determine location and shape of junctions between the main well and lateral branches, to permit the insertion of tools into such branches (Figure 1). Oil wells are filled with an opaque fluid preventing the use of any range sensors, and have extremely high pressures (up to 1500 atm) and temperatures (up to 300 ^{o}C) that make force/torque and contact sensors unreliable (Mazzini et al., 2011). This work considers the exploration of such environments by a robot equipped with only position sensors (proprioception), an approach called Whole Arm Exploration .

Methods for robotically mapping environments have largely used range sensors (Thrun, 2005). These methods are very effective, but not applicable in harsh environments. Concepts have been proposed for tactile mapping of surfaces. For example, some works distinguish an object among a library of known objects (Keren et al., 2000; Petrovskaya and Khatib, 2011). Others explore simple surfaces such as concave objects (Moll and Erdmann, 2004; Okamura and Cutkosky, 2001). However, these works cannot explore an

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_45,

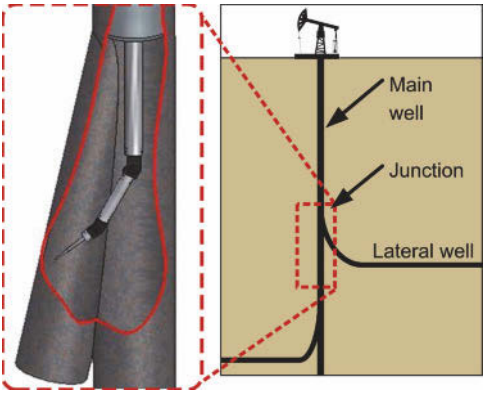


Figure 1. Tactile Mapping of the Junctions in Oil Wells

unknown environment, and they all require a force-torque or tactile sensor, which can be unreliable in harsh environments. Studies have been proposed to estimate the contact location with only position sensors when a manipulator's link is sliding against an object, and they have been used to map a simulated 2D pipeline using a snake robot (Kaneko and Tanie, 1994; Huber and Grupen, 1994; Everist and Shen, 2009). However, these methods can only be applied to a planar environment and sliding motion on the surface is time consuming and usually not feasible in a harsh environment, see (Mazzini et al., 2011). In conclusion, previously developed proprioceptive methods, while valuable contributions, cannot be used for the exploration of harsh, constraining environments.

Our recent studies have addressed the tactile exploration of unknown full 3D surfaces using a manipulator with only position sensors (Mazzini and Dubowsky, 2012; Mazzini, 2011). However, these works assume that contact occurs on the manipulator's tip, so that the contact location is known from the manipulator's kinematics. Here, this work is extended to cases where contact occurs anywhere on the manipulator. This approach is similar in spirit to Whole Arm Manipulation (Gordon and Townsend, 1989).

2 Proprioceptive Tactile Mapping

This work maps an arbitrarily-shaped, unknown environment using a manipulator mounting only position sensors. Contact with the environment is detected by monitoring the constraints on the manipulator's motions. The contact on the manipulator is not known a priori.

The following assumptions are made: the manipulator has a fixed base and its geometry is known (links are cylinders and joints are spheres); the environment is static, rigid and regular enough to be described by a combination of geometric primitives such as planes, spheres, cylinders, cones, and tori. A mesh can be used for the surface parts that cannot be well described by such primitives. The tradeoff between exploration speed and precision is controlled by a parameter, B, denoting the smallest expected surface curvature radius. The manipulator is controlled with an impedance controller so that when the manipulator touches a surface, its motion is obstructed and contact is detected. The impedance controller requires very low forces to detect contact with the environment: this insures that there is little danger to either the robot or the environment during contact.

Surface tactile mapping can be divided into two tasks: efficiently acquiring the environment data and interpreting such data to create a map of the environment. Data acquisition is a motion-planning problem that consists of choosing the manipulator's movements in order to collect the most informative data as quickly as possible. A strategy has been developed for this using an information-theoretic metric (Mazzini, 2011). This paper focuses on the data interpretation, developing a new approach to interpret contact data from a manipulator that is not restricted to end-point contact. It is composed of three steps. First, the link that is in contact is identified. Second, contact data are probabilistically classified into two subsets: contact either on the tip or on the link's body. Third, the two data subsets are used to create a model of the environment.

The link in contact is identified by monitoring joint velocities and commanded torques, as in (Kaneko and Tanie, 1994). When contact occurs at the nth link of the manipulator, all the links from 1 to n will stop, but the outer links still continue to move as long as some torque is applied to their joints. Hence, the link in contact is the last one to stop.

Once the contact link is identified, a soft classification method evaluates the probability that contact occurs on the tip (or joint, if the link is not the last) or on the link's body. This is done considering the manipulator movements at the time of contact and the partially constructed map. Consider a planar robot coming into contact with an unknown environment after a finite displacement. The contact point must be within the area swept by the manipulator during its displacement. If points on the surface are independent and have the same probability of being empty, the probability P_T of touching with the tip is the ratio between the region swept by the tip and the region swept by the whole link, or:

$$P_T = \frac{A_{swept,\,TIP}}{A_{swept,\,TIP+LINK}} \tag{1}$$

However, points are not independent in a real environment because surfaces are continuous. Here, the correlation between surface points is described by the parameter B, the smallest expected curvature radius. Constraining the range of expected surface curvatures is equivalent to dilating the manipulator with a sphere of radius B (Mazzini, 2011), and considering the area swept by the dilated robot, see Figure 2.

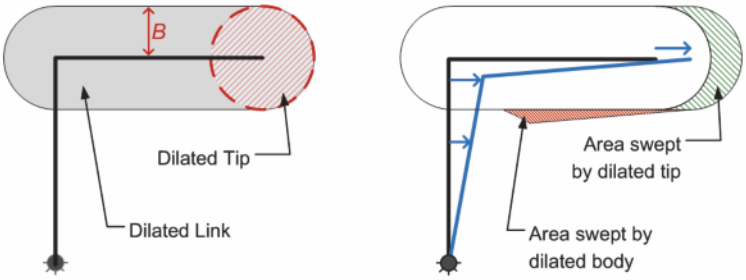


Figure 2. Dilation Area swept by dilated robot and dilated tip.

If the manipulator moves in a partially known environment, the swept area is replaced with the integral of the discrete occupancy probability map over such area (Mazzini, 2011). Here, a simple binary map that only consider the region swept by robot's movements is used. Finally, by making the manipulator displacement infinitesimal, the area swept by a point becomes the dot product between its velocity and the vector normal to the (dilated) manipulator, $\vec{n}_x \cdot \vec{v}_x$:

$$P_{TIP} = \frac{\int_{DIL,\,TIP} (\overrightarrow{n}_x \cdot \overrightarrow{v}_{TIP})\;\; P_C\,(g(x))\,dx}{\int_{DIL,\,LINK+TIP} (\overrightarrow{n}_x \cdot \overrightarrow{v}_x)\;\; P_C\,(g(x))\,dx} \tag{2}$$

where P_C the probability of a cell being occupied, and $g(x)$ is a function mapping a point x the part of the manipulator that originated it (because the occupancy probability refers to the cells swept by the manipulator). Data are classified using this equation: contact points with high or low P_T are classified as tip contact or link contact respectively.

A surface model of primitives that consists of number, types and parameters of the primitives is generated from the classified contact data. To determine the number and type of primitives, and to classify each data

point to the appropriate primitive, a segmentation method called fit and grow is used (Leonardis et al., 1995). It is based on the fitting of primitives to small initial regions and the gradual expanded or regions providing good fits. This required a method to fit both type of data (link and tip contacts) to each primitive.

Surface fitting when contact occurs on the manipulator's tip is well known (Mazzini et al., 2011); here, the same approach is extended to link contact data. If several tip contact points belong to the same primitive, its parameters can be determined with a least squares approach, minimizing the sum of the squared distances between surface and points (Figure 3). If contact is on a circular joint, or if the tip is a circle/sphere of radius w, the same least square minimization can be used by squaring, instead of the distance point-surface d, the difference between d and w. When contact occurs along the link, the contact point is not known but another constraint can be imposed: the surface must be tangent to the side of the link. A least squares minimization can be formulated by requiring the distance from the tangent line to the surface to be zero (Figure 4). Since both tip and link constraints can be expressed as a distance, they can be used together in a common minimization. To account for uncertainty in the tip/link classification process, data points are weighted by the reciprocal of the variance, an estimate of which can be obtained from the tip probability (see (Mazzini, 2011)). The use of weightings reduces the effect of wrong classification, because errors from uncertain points are discounted. Denoting by the number of "tip contacts", the number of "link contacts", the link axis, S the primitive, and the width of the link, the minimization to determine the primitive's parameters θ is:

$$\theta = \arg\min_{\theta} \left(\sum_{i=1}^{N_T} \frac{1}{\sigma_i^2} \left[d\left(P_i, S(\theta)\right) - w_i \right]^2 + \sum_{i=1}^{N_L} \frac{1}{\sigma_i^2} \left[d\left(L_i, S(\theta)\right) - w_i \right]^2 \right) \quad (3)$$

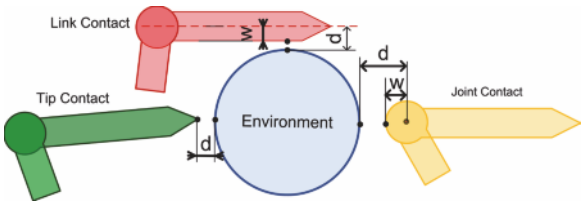


Figure 3. Least squares minimization distance (d).

3 Simulation Case Studies

The first simulation study uses a planar manipulator in a planar environment of four lines and two circles. Figure 4 shows the progress after the first contact point (left), and at the end of the exploration (right). The tip probability (shown as percentage) improves when the region around a point is explored, converging to a value close to 100% or 0%. Link contacts are correctly identified (shown as short line segments) and used for surface fitting.

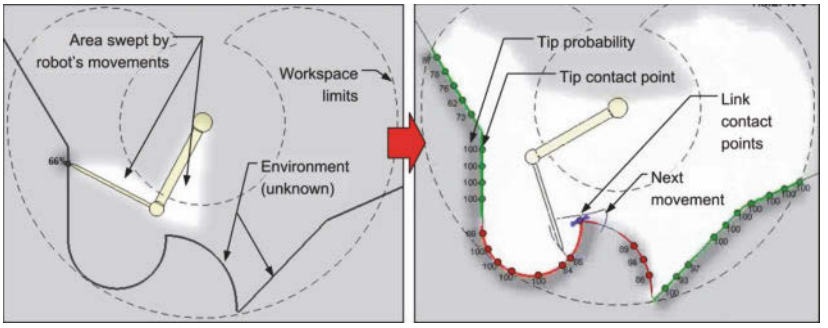


Figure 4. Exploration of a planar environment

Figure 5 shows the results of a simulation of a spatial manipulator exploring an environment composed of two cylinders, a plane and a sphere. Figure 5 shows the environment with the manipulator in its starting position (1), during the exploration (2) and the surface at the end of the exploration (3). The decrease of tip probability for one of the link contact points is shown.

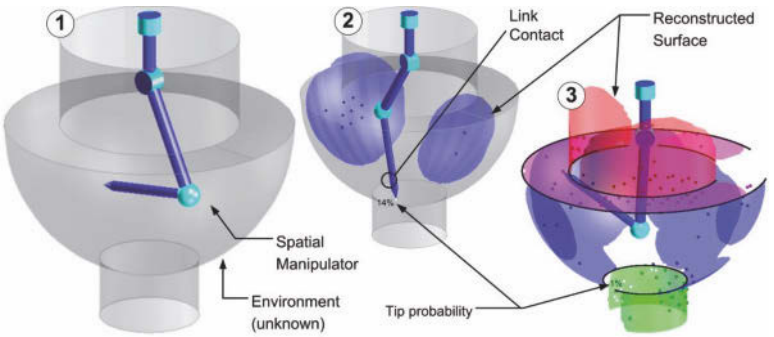


Figure 5. Spatial Exploration

4 Experimental Validation

The whole-arm exploration method has been evaluated experimentally using a two jointed planar manipulator mounting only joint encoders for sensors. Figure 6 (left) shows the manipulator with a circular tip in an environment composed of a circle and two straight lines. The manipulator first comes into contact with the circle on its link. The environment is then explored with 12 tip-contact points, and 10 link-contact points. Figure 6 shows the final reconstructed surface, superimposed with several snapshots of contact between robot and environment. The use of both types of data determines the circle radius with an error of 3%. Figure 6 (right) shows the same manipulator mounting a sharp tip while exploring an environment composed of two circles and three lines. Again the algorithm successfully explores this environment correctly recognizing both types of data.

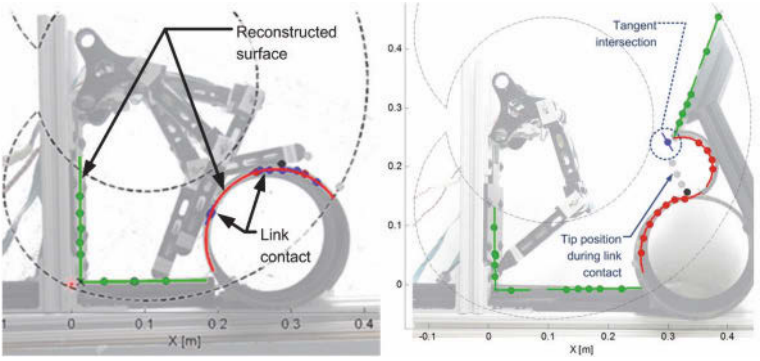


Figure 6. Experimental Validation With a Planar Manipulator.

5 Summary and Conclusions

A method to tactilely explore an unknown environment using a manipulator with only position sensors, when contact can occur anywhere on the manipulator, is presented. It consists of three steps. The first estimates what link is in contact using information from torques and velocities. The second probabilistically classifies the contact data in two subsets: tip contact and link contact. The third uses both subsets to create a model of the environment's surface. Simulations and experiments demonstrate the effectiveness of the approach.

5.1 Acknowledgements

The support of this research by the Schlumberger Doll Research Center of Cambridge MA is acknowledged.

Bibliography

J. Everist and W.M. Shen. Mapping opaque and confined environments using proprioception. In *IEEE International Conference on Robotics and Automation*, pages 1041–1046. IEEE, 2009.

S.J. Gordon and W.T. Townsend. Integration of tactile force and joint torque information in a whole-arm manipulator. In *IEEE International Conference on Robotics and Automation*, pages 464–469. IEEE, 1989.

M. Huber and R.A. Grupen. 2-d contact detection and localization using proprioceptive information. *IEEE Transactions on Robotics and Automation*, 10(1):23–33, 1994.

M. Kaneko and K. Tanie. Contact point detection for grasping an unknown object using self-posture changeability. *IEEE Transactions on Robotics and Automation*, 10(3):355–367, 1994.

D. Keren, E. Rivlin, I. Shimshoni, and I. Weiss. Recognizing 3d objects using tactile sensing and curve invariants. *Journal of Mathematical Imaging and Vision*, 12(1):5–23, 2000.

A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *International Journal of Computer Vision*, 14(3):253–277, 1995.

F Mazzini. *Tactile Mapping of Harsh, Constrained Environments, with an Application to Oil Wells.* PhD thesis, Massachusetts Institute of Technology, 2011.

F. Mazzini and S. Dubowsky. Experimental validation of the tactile exploration by a manipulator with joint backlash. *Journal of Mechanisms and Robotics*, in print, 2012.

F. Mazzini, D. Kettler, J. Guerrero, and Dubowsky S. Tactile robotic mapping of unknown surfaces, with application to oil wells. *IEEE Transactions on Instrumentation and Measurement*, (99):420–429, 2011.

M. Moll and M. Erdmann. Reconstructing the shape and motion of unknown objects with active tactile sensors. *Algorithmic Foundations of Robotics V*, pages 293–310, 2004.

A.M. Okamura and M.R. Cutkosky. Feature detection for haptic exploration with robotic fingers. *The International Journal of Robotics Research*, 20 (12):925–938, 2001.

A. Petrovskaya and O. Khatib. Global localization of objects via touch. *IEEE Transactions on Robotics*, 27(3):569–585, 2011.

S. Thrun. *Probabilistic Robotics.* the MIT Press, 2005.

BIM Based Indoor Navigation System of Hermes Mobile Robot

Barbara Siemiątkowska, Bogdan Harasymowicz-Boggio, Maciej Przybylski, Monika Różańska-Walczuk, Mateusz Wiśniowski, Michał Kowalski

* Warsaw University of Technology, The Institute of Automatic Control and Robotics, Warsaw, Poland

Abstract

In this paper the problem of BIM based indoor navigation is considered. The purpose of the project is to develop the semantic navigation system of an autonomous robot using BIM (Building Information Modeling). The described representation enables semantic robot navigation with a goal specified at a various levels of abstraction. The concept of hierarchical action planning is presented, where the plan is a time-optimized path combined with a sequence of actions required for robot movement across the whole building. The navigation process is supported by semantic localization which utilizes two methods: object detection based on point clouds (the 3D camera data acquired and converted into a point cloud) and visual object detection (based on the image taken from two color cameras placed on the sides of the robot).

1 Introduction

The aim of our project is to build the navigation system of an autonomous mobile robot which performs delivery tasks in an indoor environment. The problem of autonomous transport has been already solved in industrial environments but the adaptation of such a system to office or hospital environment is very difficult. The building adaptation process for moving autonomous platforms Fahimi (2009) (eg. appropriate markers on the floor, wide corridors, etc.) is usually very expensive and time consuming. Such robots move in partially known space which is frequented by employees and customers, so the objects can change their states M.Przybylski and B.Siemiątkowska (2012); Shiller et al. (2007). Usually the goal given to a robot is described using natural language, for example "go to the library" or "go to the kitchen". These facts have to be included in the robot motion control system Borenstein et al. (1996).

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_46,

Most of the navigation systems use a global metric map of the environment. This kind of representation is convenient for a small flat area but is inefficient in a large building. Metric maps require a lot of memory and the algorithm of path planning based on a metric map is highly time consuming. Some authors suggest using topological-metric representation Thrun et al. (2005); Harmelen et al. (2008). This kind of a map is usually represented as a graph. The nodes of the graph correspond to particular elements of the building (for example a hall, a kitchen etc.). The edges of the graph reflect accessibility relations between corresponding places. Representing an environment as a simple graph seems to be natural but it is insufficient in large buildings. In our approach a hierarchical BIM-based graph is used. *Building Information Modeling* (BIM) gives a useful tool for environment representation Siemiątkowska et al. (2011a). It is the object-based parametric modeling technique. The components of a building are described as a hierarchy of objects that have attributes and application rules. The data structure allows the extraction of multiple views of the objects.

In comparison to classical navigation system which consists of mapping, localization path planning, semantic-based navigation system consists of object recognition methods are presented in this paper) and map updating, localization and action planning. This kind of representation allows us to describe the goal in a symbolic way. The robot should be able to react to different states of objects and change the environment in order to accomplish the given task.

The mobile robot used in our project is based on the concept of modularity and susceptibility to modification. An easy access to components, which does not require time consuming disassembling, is assumed. Apart from low-level sensors used to detect obstacles around the robot, the platform is equipped with a laser range finder, two color cameras and a Kinect 3D vision sensor. Those high level sensors allows the robot semantic classification of the objects.

2 BIM-based map representation

BIM (Building Information Modeling called) is the process of generating and managing building data. This data includes inter alia, information about geometry of buildings, shapes of construction elements, dimensions of these elements and their spatial relations. Most of the buildings designed during last 30 years was created in accordance with this model. Potentially there are many buildings in which it is possible to use the presented solution.

The structure of the documentation of the building model created using BIM allows us to create many different representations useful in mobile

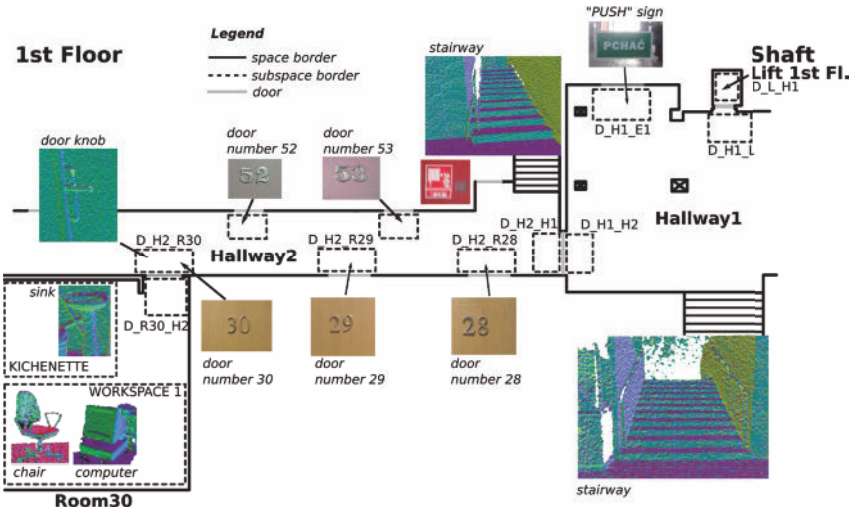


Figure 1. The map of a floor

robot navigation such as grid-based, voxel, topological, semantic or elevation maps.

The BIM-based map is a very reach space representation which provides a wide variety of qualitative information. The topology of a building is described explicitly by inclusion relation of rooms, storeys etc. The BIM-based representation can also provide an additional knowledge about functionalities of a particular place in a building. Such a representation can be easily utilized for path, task and action planning in general.

Figure 2 depicts an example of a topological map based on BIM information together with a sample plan of a path between rooms and storeys. The proposed map representation has a hierarchical structure which is a big advantage and makes it possible to employ hierarchical planning algorithms.

The hierarchy of the map is obtained directly from BIM. Four main layers are considered. The building layer is the highest one. The building consists of storeys, while storeys consist of spaces (rooms, hallways, etc.), which is natural. An additional level is the level of subspaces which may be considered as functional or operational spaces inside the spaces. The presented representation makes it possible to add even more levels such as occupancy grid which provides a smooth transition between topological and grid-based metric maps.

The pass-ability between subspaces is defined by two simple rules. It is

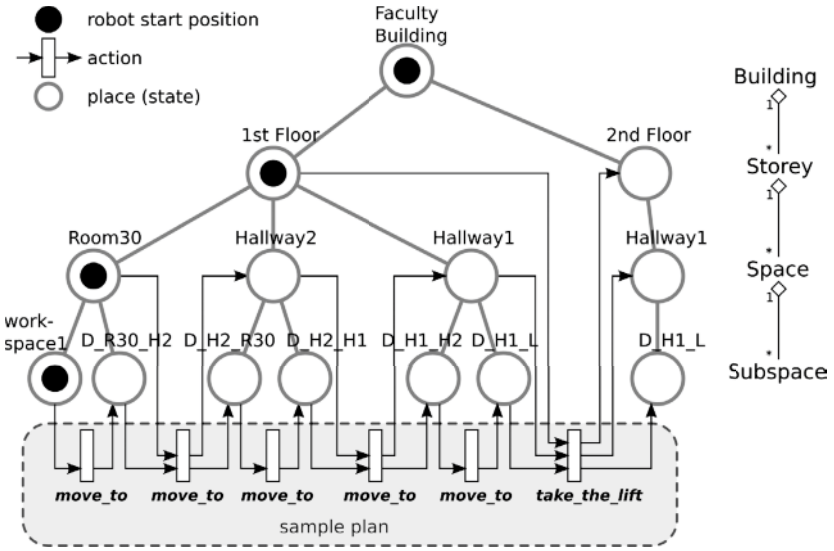


Figure 2. Hierarchical map of the Faculty Building with the sample travel plan

possible to move between two subspaces if these subspaces have the same parent (e.g. room) or if they are connected by a doorway. This information is obtained from BIM. The pass-ability property propagates in a bottom-up direction, thus moving from the $Room30$ to the $Hallway2$ is possible because of pass-ability between the D_R30_H2 and D_H2_R30 spaces. The plan execution can be represented with the use of Petri Nets Cassandras and Lafortune (2006), where firing of an action leads to transition at more than one level (e.g. $move\_to(D\_R30\_H2, D\_H2\_R30)$).

3 Hierarchical Path Planning

A typical path planning problem may be considered as a special case of an action planning, where only one action *move_to* is available. Such an approach is not sufficient for navigation across the whole building. Therefore the shortest path search has to be carried out in a state space, where transition between states is possible by applying an action Nau D. (2004).

In our approach algorithm searches for the shortest path in terms of time not space. Therefore, the action duration time, as well as the duration time of presence in a particular place (state), has to be considered.

Each action provides a cost function which calculates the transition time between two states. It is important that the symmetry of the cost function is not assumed. The cost function can be calculated even for states which are not neighbouring states, which is considered as a heuristic then. For *move_to* action it is an euclidean distance divided by the maximal allowable robot's velocity, which of course is also a heuristic. The more accurate action duration time can be obtained from the path planning algorithm working on a typical occupancy grid M.Przybylski and B.Siemiątkowska (2012).

A graph-like structure is achieved at each level of hierarchy. However, apart from cost value associated with each graph edge, the cost of presence in a node is also taken into account.

The planning procedure is carried out in an iterative top-down manner. After the calculation of the path at a higher level, the procedure is being performed at the lower level but only for places which belong to the path, which decreases search space significantly. It is important that the action duration time is the same at each level. The duration time of presence in a particular place at the particular level is the output of the sub path planning procedure being carried out at the lower level (e.g. the time of travel between $workspace1$ and D_R30_H2 determines the time of presence in the $Room30$). However, each place (space, subspace, etc.) may differ in a maximal allowable velocity and the time of presence at a place may increase after sub-path calculation. In order to find a time-optimal path, the whole planning procedure has to be carried on iteratively until convergence. The algorithm has been implemented using approach based on Cellular Neural Nets described in M.Przybylski and B.Siemiątkowska (2012), where each place (node on the graph or state in the search space) is represented by a neural cell. Each cell tries to minimize its value by choosing the best applicable action in that state. In a fact, the algorithm is an implementation of approach known as a dynamic programming Sniedovich (2006).

Figure 2 depicts a sample plan. In contrast to the typical path planning problem, the solution is no longer a simple list of places to visit. The plan consist of consecutive places and actions which have to be applied at those places in order to reach the goal. The robot placed in *workspace1* in *Room30* is asked to go to the lift which is placed in *Hallway1*. The plan is calculated at the *Space* level first. After that step the sub-problem is solved for each place along the path, which are $\{Room30, Hallway2, Hallway1\}$. If the path cost computed at the lower layer of the map is bigger than the cost assumed at the higher level, planning at higher level is repeated.

The plan presented in figure 2 is simplified to the aspect of robot movement only. However some additional action has to be carried out in order to fulfil such precondition of the *move_to* action like opened door. Therefore,

nodes placed next to the door holds additional actions applicable in that state like *open_the_door, close_the_door*. If a node is placed next to the lift then the additional action *take_the_lift* is possible to perform.

4 Semantic localization

Subspaces at the lowest level of the hierarchy presented in the previous section are nodes of the topological map used for navigation of the mobile robot. Thus semantic localization is essential.

The localization is performed based on reference frame attached to each node. The frame consists of a list of semantic features which can be observed in the corresponding subspace of an environment. The detection of a feature increases the probability of correct localization. If the probability is above a given threshold the robot assumes its presence in the subspace which is represented by the hierarchical map node.

At presented stage of our project two kinds of semantic features have been used, geometric features obtained from data collected by a 3D sensor and visual features.

Object detection from point cloud

The process of object classification is described in Siemiątkowska et al. (2011b). The data (from 3D camera) is acquired and converted into a point cloud. The point cloud is represented as RGB image. The objects are classified using SIFT features. The system is being taught (manually) and the results are added to the database. Then the main matching algorithm determines if one or more of known objects is recognized in a current view. When the robot moves along a corridor 3D metric data is not sufficient and visual information is required.

Visual object detection

The image of a known object observed from certain point can be stored as a 2D pattern. This approach is especially useful for recognition of flat objects such as door signs.

The proximity of such patterns is detected based on the image from two color cameras placed on the sides of the robot. Each node for which this 2D pattern feature is used has an attached image of a characteristic pattern which can be recognized. Some examples of patterns used to test the system are shown in figure 1.

The pattern recognition algorithm assumes that the rotation of the pattern relative to the robot is constant and the distance is within a known range. These assumptions are possible since the position of the cameras is fixed and what we want to detect is the presence of the robot within a limited area.

The pattern is recognized based on the Fast Match Template algorithm created by Tristen Georgiou Georgiou (2005). The algorithm is an improved version of the template matching method from the OpenCV library. The algorithm consists of the following steps:

1. The input image and the pattern are downscaled by a power of 2 factor;
2. The scaled input image is searched for the scaled pattern using the standard OpenCV template matching method;
3. A list of best matches (points) is created;
4. For each point of this list the original camera input image is searched for the pattern in a given area centered at the found point;
5. If the match score is above a given threshold, the pattern is considered to be found.

The results of this method are as good as the results of the simple template matching implemented in OpenCV, but the image processing time is significantly shorter.

Using this method the image from the camera is searched for a number of scaled versions of the expected pattern simultaneously, so it can be found when observed from a variable distance.

5 Experiments and Summary

This paper presents the first step toward BIM-based navigation. The problem of localization and semantic classification of certain class of objects has been solved. The localization and classification algorithms have been tested using the mobile robot. The advantages of classification based on point cloud are presented in Siemiątkowska et al. (2011b). The sample objects which are recognized by our system are presented in fig. 1. The template matching algorithm is effective and allows robot localization in the hallway. When moving parallel to the wall at a distance of 0.8–2 m the robot is able to recognize patterns in a range of about 0.5 m along the hallway. Some sample templates are presented in fig. 1. The path planning (between subspaces) has been tested in a dynamic environment with the use of the mobile robot.

In comparison to the path planning with an occupancy grid, the hierarchical method reduces the search space significantly. Moreover, the map

representation proposed in this paper allows to solve the problem of navigation between different storeys of the building.

Acknowledgments: This work has been supported by the National Science Centre (grant 2011/01/B/ST6/07385).

Bibliography

J. Borenstein, H. Everett, and L. Feng. *Where am I? Sensors and Methods for Mobile Robot Positioning*. MIT Press, 1996.

Christos G. Cassandras and Stephane Lafortune. *Introduction to Discrete Event Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

F. Fahimi. *Autonomous Robots. Modeling, Path Planning, and Control*. Springer, 2009.

T. Georgiou. Fast match template – opencv wiki, 2005. URL `http://opencv.willowgarage.com/wiki/FastMatchTemplate`. [Online; accessed 19-April-2010].

F. Harmelen, L. Lifschitz, and B. Porter, editors. *Handbook of Knowledge Representation*. Elsevier, 2008.

M.Przybylski and B.Siemiątkowska. A new cnn-based method of path planning in dynamic environment. In *The 11th International Conference on Artificial Intelligence and Soft Computing ICAISC 2012*. LNCS, Springer, 2012.

Traverso P. Nau D., Ghallab M. *Automated Planning: Theory & Practice*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004. ISBN 1558608567.

Z. Shiller, F. Large, S. Sekhavat, and C. Laugier. *Motion Planning in Dynamic Environments*. Springer, 2007.

B. Siemiątkowska, J. Szklarski, and A. Borkowski. *Towards Semantic Navigation in Mobile Robotics*, volume 5765. Springer, 2011a.

B. Siemiątkowska, J. Szklarski, and M. Gnatowski. Mobile robot navigation with the use of semantic map constructed from 3d laser range scans. *Control and Cybernetics*, 40:437–453, 2011b.

Moshe Sniedovich. Dijkstra's algorithm revisited: the dynamic programming connexion. *Control and cybernetics*, 35(3):87–92, 2006.

S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.

Overload Protection Mechanism for 6-axis Force/Torque Sensor*

Kenji Hashimoto†, Teppei Asano†, Yuki Yoshimura†, Yusuke Sugahara‡, Hun-Ok Lim§¶ and Atsuo Takanishi†¶

† Department of Modern Mechanical Engineering, Wasea University, Japan
‡ School of Science and Engineering, Kokushikan University, Japan
§ School of Engineering, Kanagawa University, Japan
¶ Humanoid Robotics Institute, Waseda University, Japan

Abstract This paper describes an overload protection mechanism for a 6-axis force/torque sensor. It can limit the load applied to the force sensor by contacting a sole part with a top plate placed on the force sensor. Specifically, compression springs are arranged between the force sensor and the top plate, and the springs are pre-compressed. When more loads than the preload are applied to the top plate, the top plate touches the sole part by compressing the springs, and it is possible to prevent the 6-axis force/torque sensor from being over-loaded. Verification of the proposed mechanism is conducted through experiments with a human-carrying biped robot, WL-16RV.

1 Introduction

Many biped robots generate motion patterns based on Zero Moment Point (ZMP) criteria, and such robots need force sensors in the feet to measure ZMP. Although there are many kinds of force sensors, especially most humanoid robots such as ASIMO (Hirose and Ogawa, 2007), HRP-2 (Kaneko et al., 2004) and WABIAN-2 (Ogura et al., 2006) have 6-axis force/torque sensors in the feet.

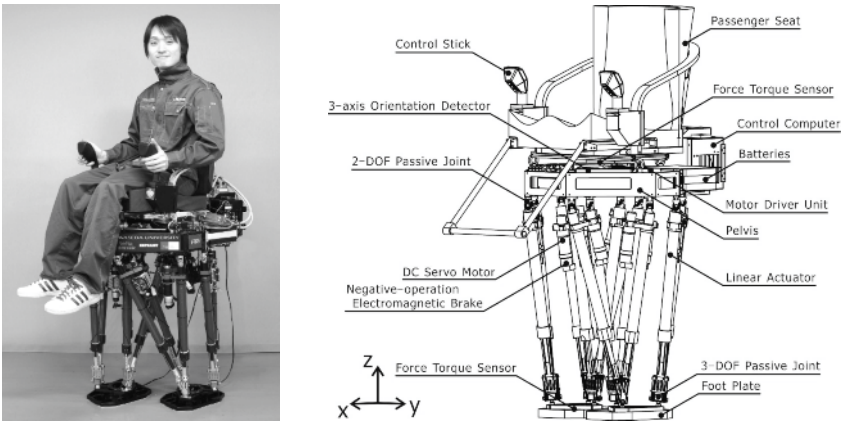
*This study was conducted as part of the Advanced Research Institute for Science and Engineering, Waseda University, and as part of the humanoid project at the Humanoid Robotics Institute, Waseda University. It was also supported in part by Global COE Program "Global Robot Academia" from the Ministry of Education, Culture, Sports, Science and Technology of Japan, TMSUK Co., Ltd., HEPHAIST Seiko Co., Ltd., and SolidWorks Japan K.K., whom we thank for their financial and technical support.

V. Padois, P. Bidaud, O. Khatib (Eds.), *Romansy 19 – Robot Design, Dynamics and Control*, CISM International Centre for Mechanical Sciences, DOI 10.1007/978-3-7091-1379-0_47,

Paying attention to commercialized 6-axis force/torque sensors, a force sensor sold by WACOH-TECH_Inc. (2012) have an overload protection mechanism. But the compression load rating is as small as 200 N, and it's difficult to use it for human-size humanoid robots due to the limitation of the compression load rating. Although JR3_Inc. (2012) sells a 6-axis force/torque sensor whose load rating is around 1000 N, it may break down if an unexpected load such as a landing-impact force added to the sensor because it does not have an overload protection mechanism.

Meanwhile, we have developed a human-carrying biped robot, Waseda Leg - No.16 Refined V (WL-16RV) that can let a human overcome the typical barriers within the human environment (Hashimoto et al., 2010) (see Figure 1). The robot consists of two legs and a waist and is capable of walking independently with an unladen weight of 75 kg. The maximum payload is 80 kg and the weight of the total system becomes 155 kg. This kind of robot requires a 6-axis force/torque sensor which has a larger load rating compared with a humanoid robot because a biped vehicle must carry a heavy load. Furthermore, there is a possibility that an unexpected force will be applied to a force sensor by load's motions during robot's walking, so a force sensor should have an overload protection mechanism.

In this paper, we develop an overload protection mechanism for a 6-axis force/torque sensor and verify the significance through experiments by WL-16RV mounted on the overload protection mechanism.



(a) (b)

Figure 1. Waseda Leg - No.16 Refined V (WL-16RV): (a) Photograph; (b) Mechanism and sensor equipped with WL-16RV.

2 Overload Protection Mechanism for 6-axis Force/Torque Sensor

2.1 Required Specification

We use a 6-axis force/torque sensor sold by JR3 Inc. for WL-16RV, and the model number is IFS-67M25T50-M40BS. The load rating is given in Table 1. The allowable load is four times as large as the load rating, and according to the sales company, a force and a torque can be linearly measured up to 200% of the load rating. When weights were actually put on the sensor, we could measure the compressive load F_z up to 1600 N and the moment loads M_x, M_y up to 60 Nm.

In this research, we set the stopper load of the overload protection mechanism as "the load which a 6-axis force/torque sensor can measure linearly" as shown in Table 1. The yaw moment M_z and the translational forces F_x and F_y are not protected because we know through preliminary experiments that these forces and moment do not exceed the load rating even if the robot carries a heavy load.

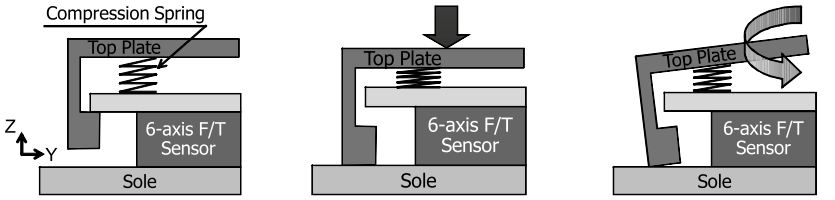
Table 1. Specification of a 6-axis force/torque sensor and a stopper load.

	F_z N	M_x, M_y Nm
Load rating	900	30
Stopper load	1600	60

2.2 Outline of Overload Protection Mechanism

An overload protection mechanism has a contraction part between a 6-axis force/torque sensor and a top plate placed over the force sensor (see Figure 2 (a)). When larger loads than a preload are applied to the top plate, the contraction part is contracted and the top plate touches the sole part. As a result, it's possible to prevent the force sensor from being over-loaded. Specifically, compression springs are arranged between the force sensor and the top plate, and the springs are pre-compressed. If suitable tension of springs is selected, the overload protection mechanism can deal with not only compressive loads but also moment loads.

When larger compressive loads than the preload are applied, the top plate touches the sole part as shown in Figure 2 (b), and it allows excessive loads to escape. When larger moment loads than the setting value are applied, the top plate touches the sole part as shown in Figure 2 (c), and it prevents the sensor from being over-loaded.



(a) (b) (c)

Figure 2. Outline of the overload protection mechanism for a 6-axis force/torque sensor: (a) Schematic view of the proposed mechanism; (b) Protection against a compressive load; (c) Protection against a moment load.

2.3 Calculation Procedure of Stopper Load

Considering stopper loads of M_x and M_y, it is ideal to arrange compression springs in a circle without any space between each spring. But in this research, we decided to place eight compression springs at regular intervals due to space limitation (see Figure 3 (a)). We can calculate the moment load M when a compression spring starts to deform as follows:

$$M = F_n \times R \sum_{n=0}^{7} \left\{ \cos\theta - \cos\left(\frac{\pi}{4}n - \theta\right) \right\} \tag{1}$$

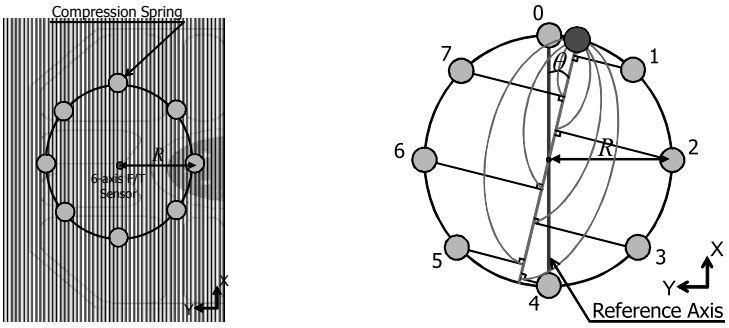
where R is the distance between the center of a 6-axis force/torque sensor and the position of compression springs ($R = 47.5$ mm), θ is the angle from the reference axis (see Figure 3 (b)), and F_n is a compressive load of the n-th compression springs ($n = 0, 1, ..., 7$). F_n is calculated as follows:

$$F_n = \frac{F_{offset} - F_z}{8} \tag{2}$$

where F_{offset} is a preload of compression springs, and F_z is a force measured by a 6-axis force/torque sensor.

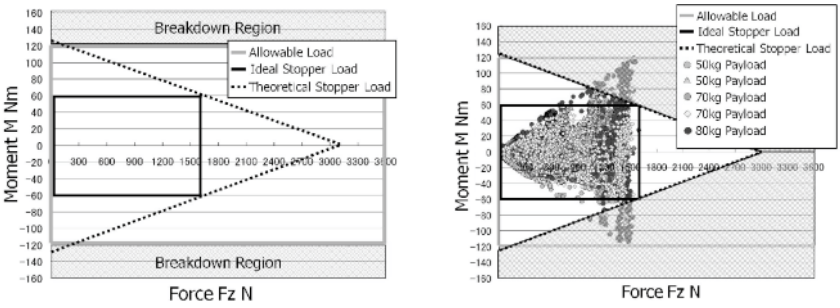
Using (1) and (2), the preload F_{offset} needed to realize the stopper load shown in Table 1 is calculated at around 3000 N. The range where the overload protection mechanism theoretically works is the black dotted line area shown in Figure 4 (a). Although an ideal is to keep loads within the black-lined area of Figure 4 (a), if only moment loads are applied, there is a possibility that larger moment loads are added to the force sensor than the allowable load of 120 Nm. But as shown in Figure 4 (b) which plots previous walking experiments' data, the relation between compressive loads

and moment loads becomes in a triangular shape, and it is unlikely that only moment loads are applied to the force sensor.



(a) (b)

Figure 3. Calculational procedure of a stopper load: (a) Arrangement of eight compression springs; (b) Definition of the variable θ.



(a) (b)

Figure 4. Relation between compressive loads and moment loads: (a) The range of theoretical stopper load; (b) Force F_z versus Moment M_x and M_y without an overload protection mechanism for 6-axis force/torque sensor.

2.4 Development of Overload Protection Mechanism

Figures 5 and 6 show the overload protection mechanism developed. Disc springs are used for compression springs, and spring guides are placed at the center of the disc springs. Oilless bushes are located above and beneath the spring guides to improve tribological property. The weight of the overload protection mechanism is 1.1 kg.

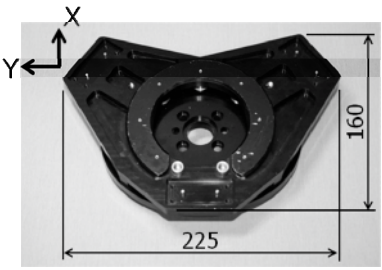
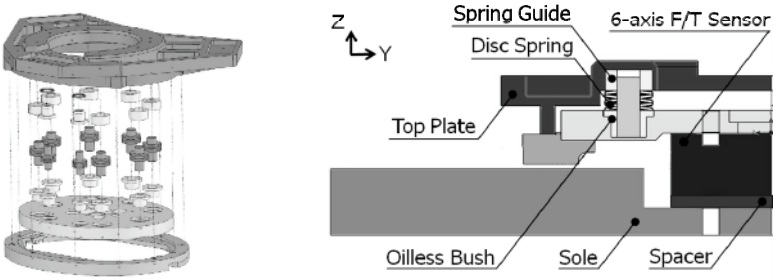


Figure 5. Overload protection mechanism for a 6-axis force/torque sensor.



(a) (b)

Figure 6. Overload protection mechanism: (a) Exploded view of the overload protection mechanism; (b) Detail view of the overload protection mechanism.

3 Experimental Tests and Consideration

We implemented the overload protection mechanism on a human-carrying biped robot WL-16RV and conducted evaluation experiments.

3.1 Evaluation Experiment with Small Stopper Load

First, we conducted walking experiments with small stopper load in order to confirm if the protection mechanism works theoretically. The weight of 30 kg and 50 kg was put on the waist of WL-16RV, and the robot walked forward with the walking cycle of 1.0 s/step and the step length of 100 mm/step and 200 mm/step respectively. As shown in Figure 7, the measuring load of a 6-axis force/torque sensor fell within the area of the theoretical stopper load. We could confirm that the overload protection mechanism worked theoretically.

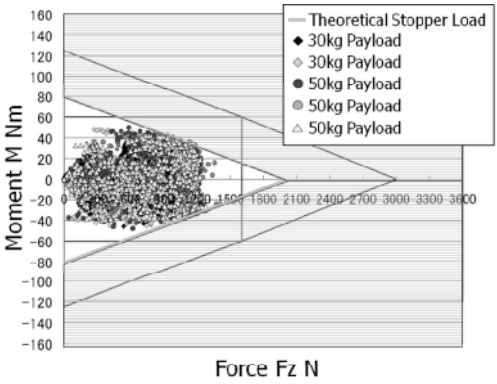


Figure 7. Measurement results with small stopper load.

3.2 Evaluation Experiment with Stopper Load of Required Specification

We conducted walking experiments with the stopper load shown in Table 1. The weight of 0 kg, 50 kg, 70 kg and 80 kg was put on WL-16RV, and the robot walked forward with the walking cycle of 1.0 s/step and the step length of 100 mm/step and 200 mm/step respectively. As shown in Figure 8, the measuring load of a 6-axis force/torque sensor were kept within the area of the theoretical stopper load, and we confirmed the effectiveness of the proposed mechanism. Although a part of moment loads is over the stopper load of 60 Nm, we consider that the protection mechanism is effective because these measuring loads are under the allowable load of 120 Nm.

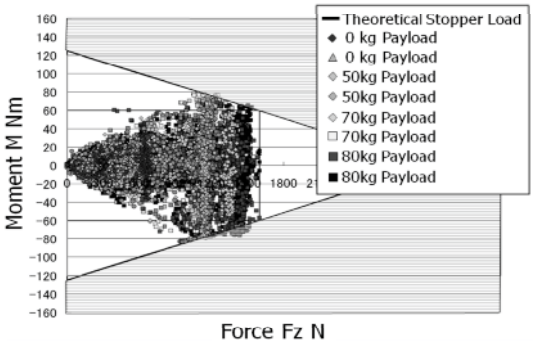


Figure 8. Measurement result with higher stopper load.

4 Conclusions and Future Work

We developed an overload protection mechanism for a 6-axis force/torque sensor mounted on biped robots. It can limit the load applied to the force sensor by contacting a sole part with a top plate placed on the force sensor physically. Specifically, compression springs are arranged between the force sensor and the top plate, and the springs are pre-compressed. When more loads than the preload are applied to the top plate, the top plate touches the sole part by compressing the springs, and it is possible to prevent the force sensor from being over-loaded. Based on the specification of the 6-axis force/torque mounted on WL-16RV, we set the stopper load of the overload protection mechanism as $F_z = 1600$ N and $M_x = M_y = 60$ Nm. The overload protection mechanism was implemented on a human-carrying biped robot WL-16RV and we conducted evaluation experiments. As a result, we succeeded to keep the loads applied to the force sensor within the area of the theoretical stopper load, and of course WL-16RV realized a stable walk while carrying a heavy load. We confirmed the effectiveness of the proposed mechanism through experiments. Although we conducted walking experiments more than 300 times until now, the 6-axis force/torque sensor has not broken down. We will continue to collect experimental data.

Bibliography

K. Hashimoto, T. Sawato, A. Hayashi, Y. Yoshimura, T. Asano, K. Hattori, Y. Sugahara, H. O. Lim, and A. Takanishi. Avoidance behavior from external forces for biped vehicle. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, pages 4715–4720, 2010.

M. Hirose and K. Ogawa. Honda humanoid robots development. *Philosophical Transactions of the Royal Society A*, 365(1850):11–19, 2007.

JR3_Inc., 2012. URL `http://www.jr3.com/`.

K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi. Humanoid robot hrp-2. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, pages 1083–1090, 2004.

Y. Ogura, K. Shimomura, H. Kondo, A. Morishima, T. Okubo, S. Momoki, H. O. Lim, and A. Takanishi. Human-like walking with knee stretched, heel-contact and toe-off motion by a humanoid robot. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3976–3981, 2006.

WACOH-TECH_Inc., 2012. URL `http://www.wacoh-tech.com/`.